

VŠB – Technická univerzita Ostrava

Fakulta elektrotechniky a informatiky

Katedra aplikované matematiky

---

## **Sekvenční Monte Carlo metody**

## **Sequential Monte Carlo Methods**

VŠB - Technická univerzita Ostrava  
Fakulta elektrotechniky a informatiky  
Katedra aplikované matematiky

## Zadání diplomové práce

Student: **Bc. Daniel Krpelík**  
Studijní program: N2647 Informační a komunikační technologie  
Studijní obor: 1103T031 Výpočetní matematika  
Téma: Sekvenční Monte Carlo metody  
Sequential Monte Carlo Methods

Zásady pro vypracování:

Skryté markovovské modely představují velice mocný nástroj pro modelování časových řad. Až na výjimky však tyto modely vedou k úlohám, které nelze řešit analyticky a je proto nutno hledat přibližná numerická řešení. K tomu jsou v praxi často využívány sekvenční Monte Carlo metody známé jako "particle filters".

Cílem práce je seznámení se se sekvenčními Monte Carlo metodami a možnostmi jejich využití při odhadování skrytých markovovských modelů a dále jejich aplikace při odhadování parametrů procesů spojených s fyzikálními ději.

Pokyny k vypracování:

1. Nastudujte základy bayesovské teorie.
2. Seznamte se s principy sekvenčních Monte Carlo metod.
3. Aplikujte sekvenční Monte Carlo metody na odhadování parametrů a predikci stavů simulovaného fyzikálního procesu při různém zatížení neurčitostí (známé parametry modelu a zašuměná pozorování; neznámé parametry modelu).

Možná rozšíření:

4. Pokuste se aplikovat tyto metody na reálná data, např. při modelování růstu trhlin v kovových materiálech.
5. Zvažte možnosti paralelizace použitých metod.

Seznam doporučené odborné literatury:

Hušková, Marie. Bayesovské metody. Univerzita Karlova, 1985

Berger, James O. Statistical decision theory and Bayesian analysis. Springer, 1985.

Doucet, Arnaud, Nando De Freitas, and Neil Gordon, eds. Sequential Monte Carlo methods in practice. Springer, 2001.


Doucet, Arnaud, and Adam M. Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. Handbook of Nonlinear Filtering 12 (2009): 656-704.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. Jan Kracík, Ph.D.**

Datum zadání: 01.09.2014

Datum odevzdání: 07.05.2015



doc. RNDr. Jiří Bouchala, Ph.D.  
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.  
děkan fakulty

Prohlašuji, že jsem diplomovou práci vypracoval samostatně. Uvedl jsem veškeré literární  
prameny a publikace, ze kterých jsem čerpal.

V Ostravě dne 7. května 2015

*Daniel Kopecký*  
.....

Chtěl bych poděkovat Ing. Janu Kracíkovi, Ph.D. za vedení při seznamování s teoretickými aspekty problému a následném vypracovávání diplomové práce.

### Abstrakt

Monte Carlo metody poskytují návod, jak převést početní problémy na problémy vyčíslení střední hodnoty náhodné veličiny. Numerická řešení spočívají v aproximaci rozdělení příslušné náhodné veličiny pro odhad hledané střední hodnoty. Jednu možnost tvorby aproximací rozdělení pravděpodobnosti poskytují Monte Carlo algoritmy. Sekvenční Monte Carlo algoritmy jsou jejich podtřídou sloužící k aproximaci sekvencí rozdělení. Takové úlohy se přirozeně vyskytují v bayesovských metodách zpracování dat. V práci se krátce zabýváme formulací bayesovských úloh a potřebou tvořit aproximace tzv. aposteriorních rozdělení. Dále uvádíme společné vlastnosti a principy sekvenčních numerických metod, základní algoritmy a jejich úpravy a následně praktické příklady jejich aplikací pro několik vybraných demonstračních problémů z oblasti bayesovské indukce.

**Klíčová slova:** sekvenční Monte Carlo, částicové filtry, bayesovská indukce, optimální filtrace

### Abstract

Monte Carlo methods provide recipe to convert numerical problems onto problems of random variable expected value estimation. Numerical solvers approximate distribution of given random variable to provide demanded estimate. Distribution approximations may be obtained by the means of Monte Carlo algorithms. Their subclass, Sequential Monte Carlo algorithms, are used for approximating sequences of distributions. Such needs arise from Bayesian data analysis. We will briefly introduce Bayesian inference and consequent need to approximate so-called aposteriori distributions. Further, we will state common properties of sequential methods, basic algorithms and their improvements and a couple of examples arising in Bayesian inference, on which we will demonstrate described algorithms.

**Keywords:** sequential Monte Carlo, particle filter, Bayesian inference, optimal filtering

# Obsah

<b>1</b>	<b>Úvod</b>	<b>5</b>
<b>2</b>	<b>Monte Carlo metody</b>	<b>7</b>
2.1	Poznámky k notaci . . . . .	7
2.2	Monte Carlo Algoritmy . . . . .	8
2.2.1	Zamítací metoda . . . . .	9
2.3	Markov Chain Monte Carlo . . . . .	10
2.3.1	Metropolisův-Hastingsův algoritmus . . . . .	10
2.3.2	Gibbsův sampler . . . . .	11
2.4	Importance sampling . . . . .	12
<b>3</b>	<b>Bayesovská statistika</b>	<b>16</b>
3.1	Bayesovská indukce . . . . .	16
3.1.1	Apriorní rozdělení . . . . .	18
3.1.2	Sekvenční odhady . . . . .	18
3.2	Skryté Markovské modely . . . . .	19
3.2.1	Optimální filtrace . . . . .	20
3.2.2	Optimální vyhlazování . . . . .	21
3.2.3	General state-space model . . . . .	22
<b>4</b>	<b>Sekvenční Monte Carlo algoritmy</b>	<b>23</b>
4.1	Particle filtery pro optimální filtraci . . . . .	23
4.1.1	Sekvenční importance sampling . . . . .	23
4.1.2	Degenerace vzorků a převzorkování . . . . .	24
4.1.3	Sekvenční Monte Carlo pro optimální filtraci . . . . .	26
4.2	Obecný sekvenční Monte Carlo algoritmus . . . . .	26
4.2.1	Transformace rozdělení . . . . .	27
4.2.2	Vytváření částic . . . . .	27
4.2.3	Pomocná sdružená rozdělení . . . . .	29
4.2.4	Odvození PF . . . . .	30
4.2.5	Zpětná jádra . . . . .	32
4.2.6	Obecný sekvenční Monte Carlo algoritmus . . . . .	33
4.2.7	Optimální vyhlazování . . . . .	34
4.3	Úpravy algoritmů . . . . .	35
4.3.1	Pomocné filtry . . . . .	35
4.3.2	Resample-move . . . . .	36
4.3.3	Move-reweight . . . . .	36

<b>5</b>	<b>Aplikace</b>	<b>38</b>
5.1	Aproximace aposteriorního rozdělení statického parametru . . . . .	38
5.1.1	Degenerovaný proces . . . . .	38
5.1.2	Nezávislá návrhová rozdělení . . . . .	40
5.1.3	Zpětná jádra . . . . .	41
5.2	Dynamické deterministické systémy . . . . .	42
5.2.1	Matematické kyvadlo . . . . .	42
5.2.2	Sledování kyvadla . . . . .	43
5.2.3	Odhad tíhového zrychlení . . . . .	45
<b>6</b>	<b>Závěr</b>	<b>49</b>
<b>A</b>	<b>Vybrané partie teorie pravděpodobnosti</b>	<b>50</b>
A.1	Vybrané definice . . . . .	50
A.2	Konvergenční věty . . . . .	51



**Seznam algoritmů**

1	Rejection Sampling . . . . .	10
2	Metropolisův-Hastingsův algoritmus . . . . .	11
3	Gibbsův sampler . . . . .	11
4	Sekvenční importance sampling . . . . .	25
5	Sekvenční Monte Carlo s adaptivním převzorkováním . . . . .	26
6	Sekvenční Monte Carlo s adaptivním převzorkováním II . . . . .	34

**Seznam obrázků**

1	Sekvenční aproximace aposteriorního rozdělení, degenerovaný proces. . . . .	40
2	Sekvenční aproximace aposteriorního rozdělení; nezávislé návrhové hustoty. . . . .	41
3	Sekvenční aproximace aposteriorního rozdělení; zpětná jádra. . . . .	43
4	Optimální filtrace výchyly kyvadla. . . . .	45
5	Aposteriovní odhad tíhového zrychlení I. . . . .	47
6	Aposteriovní odhad tíhového zrychlení II. . . . .	48

## 1 Úvod

Na Monte Carlo algoritmy můžeme pohlížet jako na numerické metody, které poskytují aproximace rozdělení pravděpodobnosti. Mnoho inženýrských problémů lze přeformulovat na problémy související s pravděpodobnostními mírami (např. vyčíslení integrálů, optimalizace, simulace ...) a k jejich řešení využít Monte Carlo algoritmů. Vzhledem k výpočetní složitosti algoritmů nastal jejich rozvoj až s vývojem počítačů. Aplikují se většinou na problémy, u kterých je složité zaručit konvergenci jiných numerických metod.

Monte Carlo algoritmy lze také přirozeně aplikovat na statistické problémy. Populární oblastí využití je aproximace aposteriorních rozdělení v bayesovské statistice. V takovém případě bývá aproximace aposteriorního rozdělení většinou mezikrok v úloze rozhodování za neurčitosti (zde spadají např. bayesovské bodové odhady). Aposteriorní rozdělení vznikající při řešení praktických úloh často nejsou explicitně vyjádřitelná v uzavřené formě a je proto nutné řešit úlohy numericky.

Monte Carlo metody teoreticky skoro jistě konvergují v nekonečném čase. I přes exponenciální nárůst výpočetních prostředků vzniká snaha navrhnout efektivnější algoritmy pro konkrétní typy problémů. Speciálně problém optimální filtrace ve skrytém markovském modelu (ekviv. sledování stavu skrytého procesu) vedl k rozvoji třídy algoritmů zvaných sekvenční Monte Carlo. Tyto efektivně využívají struktury skrytého markovského modelu pro aproximaci aposteriorních rozdělení skrytého stavu. Takové algoritmy se také označují částicové filtry [‘particle filter’] (i když jejich působnost není snadné přesně stanovit).

Úloha optimální filtrace na skrytých markovských modelech poskytuje dostatečně pružný a rozsáhlý model pro analýzu časových řad. Umožňuje nám definovat úlohu sekvenčního zpracování dat a tedy navrhovat algoritmy, které se zabývají real-time analýzou. Některé její speciální varianty (tzv. lineární Gaussovské modely) lze řešit analyticky, což vede na efektivní schémata pro konstrukci sekvenčních odhadů. Obecně ale zadaná aproximovaná rozdělení nebývají explicitně vyjadřitelná a částicové filtry jsou jednou z možností, jak problém numericky řešit (za ostatní zmiňme např. extended Kalman filter nebo variační bayesovské metody).

Vývoj sekvenčních Monte Carlo algoritmů probíhá od konkrétního (řešení problému optimální filtrace pro skryté Markovské modely [7, 8, 12]) k obecnému [5, 15]. Alternativní (i když obdobnou) třídou problému, kterou lze formulovat sekvenčně a řešit Monte Carlo algoritmy, je aproximace aposteriorního rozdělení statického parametru [4, 5]. Tato úloha, ač vypadá obdobně, se od úlohy optimální filtrace výrazně liší a je obecně považována za mnohem náročnější.

V kapitole 2 zadefinujeme obecně úlohu, kterou chceme Monte Carlo algoritmy řešit a také v jakém smyslu budeme chápat aproximace cílových rozdělení. Také zde uvedeme základní třídy Monte Carlo algoritmů, na které se budeme odkazovat později v textu. Jednak uvedeme sekvenční metody do kontextu alternativních metod, jimiž lze problémy řešit, a jednak budeme tyto základní principy využívat i při aplikaci pokročilých metod.

V kapitole 3 zmíníme základní aspekty bayesovských metod. Uvedeme, jak lze bayesovské metody využít pro formulaci úloh sekvenčního zpracování dat a několik základních úloh, které z takového postupu plynou.

V kapitole 4 se budeme věnovat sekvenčním Monte Carlo algoritmům. Nejprve uvedeme algoritmus částicový filtr pro řešení úlohy optimální filtrace na skrytém Markovském modelu. Na něm budeme demonstrovat základní principy sekvenčních metod. Sekvenční metody lze ovšem chápat i mimo kontext optimální filtrace (dokonce i mimo kontext bayesovské statistiky). Uvedeme proto obecnější

formulaci sekvenčních Monte Carlo algoritmů. Dále ukážeme, že částicové filtry, jak se vyskytují v literatuře [7, 8], umíme odvodit jako speciální případ tohoto obecného algoritmu. Uvedeme také metodu, jak řešit úlohu v této obecnější formulaci, do které spadá např. zmíněný odhad statického parametru. Doufáme, že obecná formulace nám později umožní sekvenční algoritmy dále zobecnit (např. pro bayesovské sítě nebo Markovská pole).

V kapitole 5 nakonec předvedeme sekvenční algoritmy při řešení některých vybraných problémů bayesovské indukce.

## 2 Monte Carlo metody

Historie využití stochastických numerických metod se datuje od formulace úlohy Buffonovy jehly (1777). Pokud na arch papíru rovnoměrně rozdělený rovnoběžnými přímkami házíme jehlu stejné délky jako je rozestup linek. Pravděpodobnost, že jehla linku protne je  $\frac{2}{\pi}$ . Sérii hodů můžeme považovat za Bernoulliho pokusy se známou střední hodnotou a zákon velkých čísel nám poskytne její konzistentní odhad, který můžeme využít pro vyčíslení  $\pi$ .

Úloha znázorňuje podstatu Monte Carlo (MC) metod. Přeformulujeme problém, aby matematicky odpovídal výpočtu střední hodnoty nějaké neznámé veličiny. Máme-li k dispozici soubor realizací příslušné n.v., můžeme hledanou střední hodnotu odhadnout výběrovým průměrem.

Kvalita odhadu - rozptyl - závisí na velikosti výběru. Pro zpřesnění odhadu potřebujeme vytvořit větší. V úloze Buffonovy jehly tento výběr získáváme opakováním hodů jehlou. Jelikož mnohočetné opakování fyzického experimentu a následné zpracování velkého množství dat může být časově náročné, skutečný rozvoj MC metod přišel až v polovině minulého století s příchodem počítačů.

Přispěla k tomu úloha stanovení množství neutronů, které projdou určitými látkami. Předcházející experimenty umožnili přibližně stanovit střední délky doletu neutronů a pravděpodobnost, že dojde k jejich pohlcení. Za pomoci počítačů se provedlo mnoho simulací náhodných procházek jednotlivých neutronů - realizace náhodné veličiny - a relativní množství průchozích neutronů se odhadlo výpočtem četnosti simulovaných průchodů.

Za inspiraci, která podpořila dnešní rozvoj stochastických metod, vdčíme hazardní hře ruletě. Proto se stochastické simulační metody dnes nazývají Monte Carlo algoritmy.

Více o historii řešení úlohy průchodu neutronů nalezne čtenář zde [17]. Aplikace Monte Carlo metod pro řešení inženýrských úloh je rozepsána v [14]. Další praktické aplikace Monte Carlo metod a přehled Monte Carlo algoritmů jsou uvedeny v [12].

Až v posledních letech se Monte Carlo začalo ve větší míře aplikovat i ve svém přirozeném prostředí - řešení statistických úloh. V dalších kapitolách se budeme zabývat aplikacemi v bayesovské statistice.

### 2.1 Poznámky k notaci

V práci se budeme často setkávat s náhodnými veličinami a jejich rozdělením. Budeme potřebovat pracovat jak s hustotami, tak s mírami. Abychom se neztratili ve značení zavedeme několik konvencí:

- $R(X)$  označíme obor hodnot n.v.  $X$
- $\mathcal{A}(X)$  označíme Borelovskou  $\sigma$ -algebru na  $R(X)$
- Hustoty pravděpodobnostní funkce různých náhodných veličin odlišíme argumentem, když to bude možné. V opačném případě uvedeme dolní index specifikující n.v. (tj.  $p(x) = p_X(x)$ ).
- Rozdělení ve smyslu míry odlišíme od hustoty pravděpodobnostní funkce argumentem  $dx$  (např.  $p(dx)$ ).
- Vztažnou míru na  $\mathcal{A}(X)$  označíme  $dx$  (tj.  $(R(X), \mathcal{A}(X), dx)$  je prostor s mírou a  $p(x) = \frac{dp}{dx}(x)$ ).

Dále využijeme kompaktní zápis vektoru ve formě  $a_{i:k} := (a_i, a_{i+1}, \dots, a_k)$  případně  $A^{i:k} := (A^i, A^{i+1}, \dots, A^k)$ .

Integrály uvádíme bez integračního oboru. V takových případech předpokládáme integraci přes celý nosič integrandu.

## 2.2 Monte Carlo Algoritmy

Než skutečně házet jehlou nebo pokusy jinak fyzicky realizovat, ukazuje se výhodnější experiment simulovat. K tomuto účelu slouží stochastické simulační algoritmy. Jejich smyslem je vytvořit soubory vzorků - realizací n.v. -, který můžeme využít pro odhad středních hodnot.

Výběrový soubor můžeme transformovat a využít i k odhadu střední hodnoty jiné, takto transformované, n.v.. Vzhledem k širší možnosti využití MC algoritmů je budeme dále analyzovat z pohledu aproximace funkcionálu typu:

$$I_\pi(\varphi) := \int \varphi(x) \pi(dx) = \mathbb{E}_\pi(\varphi(X)). \quad (1)$$

**Definice 2.1.** (Monte Carlo aproximace míry)

Buď  $X^{1:N}$  vzorky i.i.d. podle  $\pi(x)$   
MC aproximuje míru  $\pi(dx)$  mírou

$$\tilde{\pi}^N(dx) := \frac{1}{N} \sum_{i=1}^N \delta_{X^i}(dx),$$

kde  $\delta_a(dx)$  označuje Diracovu distribuci se středem v  $a$  ( $\int \varphi(x) \delta_a(dx) = \varphi(a)$ ).

**Definice 2.2.** (Monte Carlo odhad)

Buď  $\varphi(x)$  měřitelná funkce s konečným rozptylem,  $\pi(x)$  hustota pravděpodobnosti a  $X^{1:N}$  soubor náhodných veličin i.i.d. podle  $\pi(x)$ .

Dosažením MC aproximace míry do funkcionálu (1) získáme aproximaci

$$I_\pi(\varphi) \approx \int \varphi(x) \tilde{\pi}^N(dx).$$

MC odhad  $I_\pi(\varphi)$  označíme

$$\mathcal{E}_\pi^{MC}(\varphi(X)) := \int \varphi(x) \tilde{\pi}^N(dx) = \frac{1}{N} \sum_{i=1}^N \varphi(X^i).$$

*Poznámka 2.3.* (konzistence MC odhadu)

Platí (viz [1]):

$$\mathcal{E}_\pi^{MC}(\varphi(X)) \xrightarrow[N \rightarrow \infty]{a.s.} I_\pi(\varphi),$$

kde  $\xrightarrow[N \rightarrow \infty]{a.s.}$  označuje konvergenci skoro jistě (definice A.9).

Odhad  $\mathbb{E}_N^{MC}(\varphi)$  je nestranný a platí CLV (věta A.12) s asymptotickým rozptylem  $\mathcal{O}(\frac{1}{N})$ :

$$\frac{1}{N} \int (\varphi(x) - I_\pi(\varphi))^2 \pi(dx).$$

**Důsledek 2.4.** (*konvergence MC aproximace*)

$$\tilde{\pi}^N \Rightarrow \pi,$$

kde ‘ $\Rightarrow$ ’ označíme konvergenci v distribuci (definice A.11).

K vytváření simulovaných výběrů se většinou využívají generátory náhodných čísel (existují i postupy, které fungují na deterministickém generování výběrů, těmi se ale zabývat nebudeme). Základní generátory ale poskytují pouze omezené možnosti. Nativně lze tvořit výběry z uniformního rozdělení, které lze přímo transformovat na výběry z několika málo jiných rozdělení. Další možností je využít transformaci pomocí inverzní distribuční funkce. Pro některá rozdělení je ale tvorba výběru těmito metodami nemožná nebo neefektivní.

Monte Carlo algoritmy představují třídu metod, které nám umožní tvořit simulované výběry i se slabými požadavky na tvar cílového rozdělení. Na vytváření takových výběrů se budeme odkazovat jako na tvorbu vzorků [‘sample’] v souladu s MC terminologií. Společným požadavkem metod je možnost vyčíslení hustoty cílového rozdělení  $\pi(x)$ , ze které chceme tvořit vzorky, bodově až na normalizační konstantu.

$$\pi(x) = \frac{\gamma(x)}{Z}, \quad (2)$$

kde  $\gamma(x) \in L^1(R(X))$ ;  $\gamma(x) \geq 0$  je známá nevlastní hustota, a  $Z \in \mathbb{R}$  je normalizační konstanta

$$Z = \int \gamma(x) dx. \quad (3)$$

Normalizační konstantu  $Z$  totiž obvykle vyčíslit neumíme. Naštěstí ji algoritmy MCMC a IS nevyžadují. Stačí umět bodově vyčíslit funkci  $\gamma(x)$ .

### 2.2.1 Zamítací metoda

Zamítací metodu [‘rejection sampling’] zmiňujeme kvůli historické perspektivě. Můžeme ji využít pro vytváření souboru vzorků  $X^{1:N}$  i.i.d. podle  $\pi(x)$ . Algoritmus využívá pomocné hustoty  $q(x)$  a konstantu  $M < \infty$ . Pro tyto musí platit

$$\forall x : \pi(x) \leq Mq(x).$$

Soubor vzorků vytváříme iteračně, ajk je ukázáno v algoritmu 1.

Takto tvořené vzorky  $X^{1:N}$  budou i.i.d. podle  $\pi(x)$  [18]. Nevýhodou této metody je nutnost stanovení omezující konstanty  $M$ . V případě, že se  $\pi(x)$  a  $q(x)$  “moc liší”<sup>1</sup>, dojde k častému zahazování vzorků, což je plýtvání výpočetním časem. V případě vysokorozměrných rozdělení bývá rozdíl natolik významný, že se metoda stává neefektivní.

<sup>1</sup>Podobně vágních formulací odlišnosti dvou rozdělení se v textu budeme dopouštět vícekrát. Apelujeme na čtenářovu intuitivní představu, jelikož koncepce podobnosti by práci nic podstatného nepřinesla v porovnání s náročností jejího uvedení.

---

**Algoritmus 1** Rejection Sampling

---

**i=1;****OPAKUJ** dokud  $i \leq N$ **Vygenerujeme**  $X^{(i)} \sim q(x)$  a  $U \sim \mathcal{U}(0, 1)$  (kde  $\mathcal{U}(a, b)$  označíme uniformní rozdělení na intervalu  $\langle a, b \rangle$ )**Je-li**  $U < \frac{\pi(X^{(i)})}{Mq(X^{(i)})}$ ,**pak** vzorek přijmeme ( $X^i = X^{(i)}$ ) a navýšíme  $i = i + 1$ **jinak** vzorek zahodíme

---

**2.3 Markov Chain Monte Carlo**

Markov Chain Monte Carlo (MCMC) zaštiťuje větší třídu algoritmů [1, 19], z nichž historicky nejstarším je Metropolisův algoritmus. Algoritmy vytváří soubor závislých vzorků takovým způsobem, aby byl asymptoticky rozdělený podle  $\pi(x)$ . Principem je vytvářet vzorky z Markovského řetězce s přechodovým jádrem  $K$  se stabilním rozdělením  $\pi(x)$ .

$$\int K(x, x')\pi(x)dx = \pi(x').$$

Na takto generované vzorky už nemůžeme, kvůli závislosti, přímo uplatnit zákon velkých čísel. Rozumným požadavkem na konvergenci je ovšem konvergence nezávisle na počátečním prvku. Další podmínky stanoví teorie Markovských procesů. Často využívanou postačující podmínkou konvergence odhadu je tzv. zpětná reverzibilita jádra aneb.

$$\forall x, x' : \quad \pi(x)K(x, x') = \pi(x')K(x', x)$$

**2.3.1 Metropolisův-Hastingsův algoritmus**

Metropolisův-Hastingsův algoritmus byl prohlášen za jeden z nejvýznamnějších algoritmů století. Vznikl zobecněním Metropolisova algoritmu, který zmíníme později jako speciální případ.

V obecném kroku  $k$ , kdy máme k dispozici vzorek  $X^{k-1}$  vygenerujeme nový vzorek  $X' \sim q(x'|X^{k-1})$ , který přijmeme ( $X^k \leftarrow X'$ ) s pravděpodobností  $A(x^{k-1}, x') = \min \left\{ 1, \frac{\pi(x')q(x^{k-1}|x')}{\pi(x^{k-1})q(x'|x^{k-1})} \right\}$ . V opačném případě opíšeme předchozí vzorek  $X^k \leftarrow X^{k-1}$ .

Přechodové jádro Markovského procesu v Metropolisově-Hastingsově algoritmu má tvar

$$K(x, dx') = q(dx'|x)A(x, x') + \delta_x(dx')r(x), \quad (4)$$

kde člen  $r(x)$  je spjat s odmítnutím vzorku

$$r(x) = \int q(x'|x)(1 - A(x, x'))dx.$$



---

**Algoritmus 2** Metropolisův-Hastingsův algoritmus

---

 $X^0 \sim q_0(x);$ **Pro**  $k = [1, N]$ **Vygenerujeme**  $X' \sim q(x'|X^{k-1})$  a  $U \sim \mathcal{U}(0, 1)$ **Je-li**  $U < A(x^{k-1}, x') = \min \left\{ 1, \frac{\pi(x')q(x^{k-1}|x')}{\pi(x^{k-1})q(x'|x^{k-1})} \right\},$ **pak** vzorek přijmeme ( $X^k \leftarrow X'$ )**jinak**  $X^k \leftarrow X^{k-1}$  a nový vzorek zahodíme

---

Tento řetězec splňuje postačující podmínku konvergence.

Metropolisův algoritmus můžeme popsat jako speciální případ Metropolisova-Hastingsova algoritmu, při volbě symetrických návrhových hustot  $q(x'|x)$  (tj.  $q(x|x') = q(x'|x)$ ). Pravděpodobnost přijetí se pak redukuje do podoby  $A(x, x') = \min \left\{ 1, \frac{\pi(x')}{\pi(x)} \right\}$ .

**2.3.2 Gibbsův sampler**

Gibbsův sampler aplikuje obecný (nejen) matematický princip rozkladu úlohy do posloupnosti jednodušších podúloh. Slouží pro generování vzorků z vysokorozměrných rozdělání.

Požadavkem metody je znalost tzv. úplných podmíněných rozdělání  $\pi(x_k | x_{1:k-1}, x_{k+1:d}); \forall k$ . Vytváření vzorku  $(X_1, X_2, \dots, X_d)^i$  pak můžeme rozdělit do více kroků tak, že pro všechna  $k \in [1, d]$  zvolíme návrhové rozdělání:

$$q(x'|x) = \begin{cases} \pi(x'_k | x_{-k}) & ; \text{pokud } x'_{-k} = x_{-k} \\ 0 & ; \text{jinak} \end{cases},$$

kde  $x_{-k} := x_{1:k-1}, x_{k+1:d}$ .

Pravděpodobnost přijetí v případě Gibbsova sampleru je  $A(x, x') = 1$  [1]. Přijímáme tedy každý vzorek.

Vzorkování z úplného podmíněného rozdělání řešíme jako samostatnou podúlohu. Výhodou tohoto postupu je, že v mnoha případech má úplné podmíněné rozdělání jednodušší tvar, např. tvoří konjugované systémy. V opačném případě můžeme využít jiné MC algoritmy (např. MCMC).

---

**Algoritmus 3** Gibbsův sampler

---

 $X_{1:d}^0 \sim q_0(x_{1:d});$ **Pro**  $i = [1, N]$ **Pro**  $k = [1, d]$ **Vygenerujeme**  $X_k^i \sim q(x^i | X_{1:k-1}^i, X_{k+1:d}^{i-1})$ 

---

## 2.4 Importance sampling

Metoda Importance Sampling (IS) je inspirovaná možností převodu výpočtu střední hodnoty n.v. na výpočet střední hodnoty jiné n.v. podle Radonovy-Nikodýmovy věty (věta A.6).

Místo složitěho rozdělení  $\pi(x)$  vytvoříme MC aproximaci hledaného rozdělení pomocí souboru vzorků rozdělených podle pomocné hustoty  $q(x)$ , kterou budeme dále nazývat návrhová [‘proposal’] hustota.

Je-li míra  $\pi(dx)$  absolutně spojitá vůči míře  $q(dx)$ , můžeme aproximovaný funkcionál (1) upravit:

$$\begin{aligned} I_\pi(\varphi) &= \int \varphi(x) \pi(dx) = \int \varphi(x) \frac{d\pi}{dq}(x) q(dx) \\ &= \mathbb{E}_q \left( \varphi(X) \frac{d\pi}{dq}(X) \right). \end{aligned}$$

Nyní potřebujeme zvolit vhodná návrhová rozdělení  $q(dx)$ , pro která Radonova-Nikodýmova derivace existuje a umíme ji až na normalizační konstantu vyčíslit s.v.v  $q(dx)$ .

*Poznámka 2.5.* (Derivace měr s hustotou)

Jsou-li  $p, q$  hustoty pravděpodobnosti a  $\text{supp}(p) \subset \text{supp}(q)$ , pak:

- míra  $p(dx)$  je absolutně spojitá vůči míře  $q(dx)$
- $\frac{dp}{dq}(x) = \frac{p(x)}{q(x)}$

Připomeneme předpokládaný tvar aproximované hustoty (2). Zvolme dále libovolnou návrhovou hustotu  $q(x)$ , aby  $\text{supp}(\pi) \subset \text{supp}(q)$ . Pak existuje Radonova-Nikodýmova derivace

$$\frac{d\pi}{dq}(x) = \frac{\gamma(x)}{Zq(x)}$$

a odhadovaný funkcionál můžeme vyjádřit ve tvaru

$$I_\pi(\varphi) = \frac{\int \varphi(x) \frac{\gamma(x)}{q(x)} q(dx)}{Z}$$

kde  $\frac{\gamma(x)}{q(x)}$  označíme  $w(x)$  a nazveme váhovou funkcí [‘weight function’].

Ve většině případů je normovací konstanta neznámá a proto odhad funkcionálu převedeme na podíl dvou odhadů

$$I_\pi(\varphi) = \frac{\int \varphi(x) w(x) q(dx)}{\int w(x) q(x) dx} = \frac{\mathbb{E}_q(\varphi(X) w(X))}{\mathbb{E}_q(w(X))},$$

což způsobí vychýlení výsledného odhadu  $I_\pi(\varphi)$ .

**Definice 2.6.** (Importance Sampling aproximace)

Bud'  $X^{1:N}$  vzorky i.i.d. podle  $q(x)$ ;  $\text{supp}(\pi) \subset \text{supp}(q)$ .  
IS aproximuje míru  $\pi(dx)$  mírou

$$\hat{\pi}^N(dx) := \sum_{i=1}^N W^i \delta_{X^i}(dx),$$

kde  $W^{1:N}$  je pomocný vektor vah, které splňují

$$W^i = \frac{w(X^i)}{\sum_{k=1}^N w(X^k)} \propto \frac{\gamma(X^i)}{q(X^i)}.$$

**Definice 2.7.** (Importance Sampling odhad)

Bud'  $\varphi(x)$  měřitelná funkce s konečným rozptylem,  $\pi(x), q(x)$  hustoty pravděpodobnosti a  $X^{1:N}$  soubor náhodných veličin i.i.d. podle  $q(x)$  a vektor  $W^i$  vektor vah  $W^i \propto \frac{\pi(x)}{q(x)}$ .  
Dosazením MC aproximace míry do funkcionálu (1) získáme aproximaci

$$I_\pi(\varphi) \approx \int \varphi(x) \hat{\pi}^N(dx).$$

IS odhad  $I_\pi(\varphi)$  označíme

$$\mathcal{E}_\pi^{IS}(\varphi(X)) := \int \varphi(x) \hat{\pi}^N(dx) = \sum_{i=1}^N W^i \varphi(X^i)$$

a odhad normovací konstanty

$$\hat{Z} := \mathcal{E}_q^{MC}(w(X)) = \frac{1}{N} \sum_{i=1}^N w(X^i)$$

*Poznámka 2.8.* (konzistence IS odhadu)

Platí ([8]):

$$\begin{cases} \mathcal{E}_\pi^{IS}(\varphi(X)) \xrightarrow[N \rightarrow \infty]{a.s.} I_\pi(\varphi) \\ \hat{Z} \xrightarrow[N \rightarrow \infty]{a.s.} Z \end{cases}$$

Odhad  $\hat{Z}$  je nestranný s relativním rozptylem  $\mathcal{O}(\frac{1}{N})$ :

$$\frac{D[\hat{Z}]}{Z^2} = \frac{1}{N} \left( \int \frac{\pi^2(x)}{q(x)} dx - 1 \right).$$

Odhad  $\mathbb{E}_\pi^{IS}(\varphi)$  je vychýlený s asymptotickou výchyldou  $\mathcal{O}(\frac{1}{N})$ :

$$\lim_{N \rightarrow \infty} N \mathbb{E}_q [(\mathcal{E}_\pi^{IS}(\varphi(X)) - I_\pi(\varphi))] = - \int \frac{\pi^2(x)}{q(x)} (\varphi(x) - I_\pi(\varphi)) dx.$$

Pro odhad  $\mathbb{E}_\pi^{IS}(\varphi)$  platí CLV s asymptotickým rozptylem  $\mathcal{O}(\frac{1}{N})$ :

$$\frac{1}{N} \int \frac{\pi^2(x)}{q(x)} (\varphi(x) - I_\pi(\varphi))^2 dx.$$

**Důsledek 2.9.** (*konvergence IS aproximace*)

$$\hat{\pi}^N \Rightarrow \pi.$$

Návrhové rozdělení můžeme volit téměř libovolně. Může se jednat o rozdělení, ze kterého umíme snadno vzorkovat nebo rozdělení které minimalizuje výsledný rozptyl odhadu.

V případě, kdy je normovací konstanta známá, lze ověřit, že ideální volba návrhového rozdělení  $q(dx)$  (ve smyslu minimalizace rozptylu odhadu funkcionálu  $I_\pi(\varphi)$  :

$$D_q[\varphi(X)w(X)] = \mathbb{E}_q(\varphi^2(X)w^2(X)) - I_\pi^2(\varphi)$$

) je hustota  $q^*(x) \propto |\varphi(x)| \pi(x)$  [1]. V praxi je často nemožné vzorkovat z takového rozdělení. Ukazuje nám ovšem, že je výhodnější vzorkovat v důležitých oblastech, kde je hodnota  $|\varphi(x)| \pi(x)$  vysoká. Pro pevná  $\varphi(x)$  je také možné najít rozdělení, pro která bude mít odhad  $\mathcal{E}_N^{IS}(\varphi(X))$  menší rozptyl, než odhad  $\mathcal{E}_N^{MC}(\varphi(X))$ . Proto se IS často využívá při zkoumání řídkých jevů.

Nás zajímají situace, kdy potřebujeme vyčíslit funkcionál  $I_\pi$  pro různé funkce  $\varphi(x)$  a návrhovou hustotu  $q(x)$  proto volíme “blízkou”  $\pi(x)$ . V případě neznámé normovací konstanty je rozptyl odhadu  $I_\pi(\varphi)$  zhruba úměrný  $1 + D_q[w(X)]$  [4, 12]. Pokud nás tedy zajímá aproximace funkcionálu (1), budeme se snažit volit návrhová rozdělení s co nejmenším rozptylem vah (optimální návrhové rozdělení  $q(x) = \pi(x)$  s rozptylem  $D_q[w(X)] = 0$  volit nemůžeme, jelikož nemožnost vzorkování z  $\pi(x)$  je důvodem, proč IS provádíme).

S váženými vzorky budeme dále pracovat často a zmíněný způsob, jak počítat váhy, se v dalších kapitolách ukáže příliš restriktivní. Zavedme proto následovně (viz [7, 15]):

**Definice 2.10.** (řádně vážené vzorky)

Soubor  $N$  vzorků  $X^i \sim q(x)$  a vektor vah  $W^{1:N}$  společně nazveme souborem řádně vážený vzorků podle/příslušných rozdělení  $\pi$ , pokud pro každou  $g \in L^2(R(X))$

$$\mathbb{E}_q(g(X^i)W^i) = c\mathbb{E}_\pi(g(X^i))$$

pro nějakou konstantu  $c$  a všechny vzorky  $X^i \sim q(x)$ .

Budeme značit  $\{W^i, X^i\} \sim \pi(x)$  (příp.  $\{W^i, X^i\} \sim \pi(x); X \sim q(x)$ , zajímá-li nás v následující analýze i návrhové rozdělení).

(úmluva : v textu budeme vynechávat informaci o počtu vzorků, všude nám stačí tiše předpokládat, že jich je  $N$ ).

### 3 Bayesovská statistika

Statistika je odvětvím zpracování dat. Nejprve musíme najít matematický popis, který nám umožní s daty pracovat. Budeme na ně pohlížet jako na realizace náhodné veličiny a při práci s nimi využijeme poznatků teorie pravděpodobnosti. Musíme tedy předpokládat, že existuje rozdělení, ze kterého data pocházejí (tímto začínáme modelovat).

Např. v případě fyzikálních měření se např. často užívá předpokladu, který podporují naše empirické zkušenosti, že jsou rozdělena normálně. Ve statistických metodách s nimi pracujeme jako s náhodnou veličinou  $Y \sim \mathcal{N}(y; m, \sigma^2)$ -kde  $m, \sigma^2 \in \mathbb{R}$  mohou být jak známé (např. z předchozích analýz) nebo mohou být samy předmětem odhadu.

Předpokládejme dále pevný model dat (někdy se na ně budeme odkazovat jako na pozorování)  $Y \sim l(y; x)$ , kde  $x$  je nějaký parametrem rozdělení, ze kterého pocházejí. Funkce  $l(y; x)$  se ve statistické literatuře často nazývá věrohodnostní [‘likelihood’] funkce. Tímto předpokladem pro pozorování provádíme v modelu n.v.  $Y$  s parametry  $x$ . Vystává otázka jak pracovat s neznámým parametrem.

V kontextu bayesovských metod chápeme neznáme parametry jako náhodné veličiny. Místo  $x \in \mathbb{R}$  (jak bývá obvyklé pro neznámou hodnotu) budeme neznámý parametr reprezentovat n.v.  $X \sim p(x)$ . Rozdělení  $p : \mathcal{A} \subset 2^{R(X)} \rightarrow \langle 0, 1 \rangle$ , které pro  $x$  takto vytvoříme, interpretujeme jako důvěryhodnost [‘plausibility’], že skutečná hodnota parametru (ozn.  $x_{real}$ ) leží v  $p(A)$  (např.  $p(\langle 0, 1 \rangle) = 1$  značí, že  $x_{real}$  leží s jistotou v intervalu  $\langle 0, 1 \rangle$ ).

#### 3.1 Bayesovská indukce

Nyní můžeme využít poznatků teorie pravděpodobnosti a úlohu chápat jako problém nalezení podmíněného rozdělení  $p(x|y)$ , podmíněného rozdělení n.v.  $X$  v závislosti na n.v.  $Y$  - pozorovaných datech.

Pro vyjádření podmíněného rozdělení využijeme Bayesův vzorec (odtud název pro celé odvětví - bayesovská statistika; připomněme navíc notaci  $p(x) = p_X(x)$ , kdy dolní index vynecháváme, pokud je z argumentu rozdělení zřejmé, o jakou n.v. se jedná):

$$p(x|y) = \frac{p(x, y)}{p(y)} = \frac{p_{Y|X}(y|x)}{\int p_{Y|X}(y|\xi)p_X(\xi)d\xi}p_X(x) \quad (5)$$

Vzorec nyní rozebereme po částech:

- $p(x|y)$  je hledaná podmíněná hustota, v bayesovském kontextu ji nazýváme aposteriorní hustotou
- $p(y|x)$  je hustota n.v. reprezentující pozorovaná data a je dána modelem
- $\int p_{Y|X}(y|\xi)p_X(\xi)d\xi$  je reálná konstanta (nezávisí na  $x$ ) a má význam normalizační konstanty
- $p(x)$  nazveme tzv. apriorní rozdělení  $X$ , které reprezentuje naši znalost  $x$  před zahájením experimentu

Normalizační konstanta je jednoznačně určena požadavkem  $\|p(x|y)\|_{L^1(R(X))} = 1$ , proto se kvůli zpřehlednění někdy omezíme jen na proporcionální závislosti, tedy:

$$p(x|y) \propto p(y|x)p(x).$$

Apriorní rozdělení je kontroverzním prvkem bayesovských metod. Vědecká metoda vyžaduje reprodukovatelnost výsledků. Apriorní rozdělení modeluje naše předběžné “představy”, které mohou být závislé na subjektivních úsudcích (lišících se u různých vědců a tedy při různých analýzách dat).

Bayesovská statistika nám umožní formulovat problémy související s odhady neznámých veličin. Mnoho těchto problémů vede na úlohy rozhodování za neurčitosti - optimalizační úlohy pro funkcionály obsahující n.v..

Obecná formulace rozhodovací úlohy přesahuje rámec práce, čtenáře odkážeme na [2, 13]. Omezíme se na případ, kdy je rozhodovací úloha definovaná jako nalezení nějakého minimálního rozhodnutí  $\hat{a} \in A$ :

$$\hat{a} \in \operatorname{argmin}_{a \in A} \int L(a, x)p(x)dx = \operatorname{argmin}_{a \in A} \mathbb{E}_p(L(a, X)),$$

kde  $L(a, x) : A \times R(X) \rightarrow \mathbb{R}$  nazveme ztrátovou funkcí. Úlohu můžeme interpretovat jako nalezení rozhodnutí, které minimalizuje očekávanou ztrátu. Ztrátová funkce  $L(a, x)$  je parametrem rozhodovací úlohy.

Například bayesovské bodové odhady lze formulovat jako minimalizaci střední kvadratické ztráty pro aposteriorní rozdělení  $X \sim p(x)$ :

$$\hat{x} = \operatorname{argmin}_{a \in A} \int (x - a)^2 p(x)dx = \mathbb{E}_p(X).$$

Pro nás významným aspektem je, že minimalizovaný výraz můžeme vyjádřit jako střední hodnotu transformované n.v. a pro její vyčíslení v různých bodech  $a$  můžeme využít MC aproximaci  $\hat{p}_N \Rightarrow p$ .

V kontextu rozhodování za neurčitosti můžeme formulovat i úlohu testování hypotéz, kdy ztrátovou funkci interpretujeme jako penalizaci v případě chyby I., II. druhu.

Důslednější rozbor podstaty bayesovské indukce a její uplatnění ve vědecké metodě nalezne čtenář v [11]. V textu se dále zaměříme pouze na konkrétní aspekty bayesovské statistiky související s numerickým řešením úloh. Pro ucelený popis bayesovských metod odkazujeme čtenáře na [13].

Úvahy doposud předpokládaly znalost modelu  $p(y|x)$ , což nemusí být splněno. V případě, kdy model neznáme, můžeme opět využít bayesovské metody a samotný model pokládat za kategoričnou n.v.  $M$  a odhadovat

$$p(m|y) = p(m) \frac{\int p(y|x, m)p(x|m)dx}{\int p_{Y|XM}(y|x, \eta)p_{XM}(x, \eta)dx d\eta},$$

kde  $p(m)$  je apriorní hustota na množině modelů,  $p(x|m)$  apriorní hustota  $X$  dána modelem  $m$  a  $p(y|x, m)$  věrohodnostní funkce určená modelem  $m$ .

I v případě, kdy je model zadán, bývá zvykem ověřit, jestli skutečně popisuje analyzovaná data. Tento krok se nazývá verifikace modelu a bývá součástí všech statistických metod (např. test normality dat v případě využití metody ANOVA). Význam tohoto kroku je demonstrován např. v [11].

### 3.1.1 Apriorní rozdělení

Prvním problémem formulace bayesovské úlohy je stanovení apriorního rozdělení. Z tvaru aposteriorního rozdělení (5) je patrné, že volba ovlivní řešení. Zde můžeme do analýzy dat zavést neobjektivní informaci.

Uvědomme si ale, že v každé statistické úloze je vložena apriorní informace (už samotný model je informace, kterou do analýzy vkládáme).

Může se stát, že z první analýzy dat vyplývají absurdní závěry (ve značném kontrastu s jinými empirickými daty). Prozkoumání kvality těchto závěrů nás může nasměrovat k formulaci nové úlohy s přidáním apriorní informace, která danou kontradikci vylučuje. Některé složitější úlohy se bez tohoto kroku neobejdou. Jejich podoba většinou vyplývá z post-analýzy - verifikace modelu.

Volba apriorního rozdělení není jednoznačně řešitelný problém, což může vést k rozdílům v závěrech plynoucích z dat. Pro některé úlohy platí věta Bernstein-von Mises [10], či její alternativy, která nám zaručí konvergenci vzhledem počtu pozorování (velikosti datového souboru) nezávisle na apriorním rozdělení. Tvzení ovšem neplatí pro všechny úlohy, proto musíme apriorní rozdělení volit s rozvahou.

Pokud to jde, volí se ve tvaru, aby společně s  $l(y; x)$  tvořilo tzv. konjugovaný systém - apriorní a aposteriorní rozdělení budou ležet ve stejném systému a lišit se parametry. Tato volba nám následně umožní analyticky určit podobu aposteriorního rozdělení pomocí přepočítání parametrů daného systému. Apriorní informaci v tomto případě vložíme do parametrů apriorní hustoty.

My se budeme zabývat případy, kdy takový systém vytvořit neumíme a musíme se tedy uchýlit k numerickým metodám. Existuje mnoho technik volby apriorní hustoty, více lze nalézt v [2, 13]. Častým principem je snaha přinést do analýzy minimum neobjektivní informace. Hovoříme pak o tzv. neinformativních prior. Příkladem jsou:

- Uniformní prior  $p(x) \propto 1$  - v tomto případě dostaneme  $p(x|y) \propto p(y|x) = l(y; x)$
- Jeffrey's prior  $p(x) = \sqrt{|\mathcal{I}(x)|}$ ; kde  $\mathcal{I}_{ij}(x) = \int \frac{l'_i(y; x) l'_j(y; x)}{l^2(y; x)} p(y|x) dy$  je tzv. Fischerova informační matice

### 3.1.2 Sekvenční odhady

Bude nás zajímat jiný aspekt apriorní informace. Představme si, že máme větší počet dat. Rozdělme je do disjunktních skupin náhodných veličin (resp. náhodných vektorů)  $Y_1, Y_2$ . Známe-li modely  $p(y_1|x), p(y_2|x)$ , jak bude vypadat aposteriorní rozdělení  $p(x|y_1, y_2)$ ?

$$p(x|y_1, y_2) = \frac{p(y_1, y_2|x)}{p(y_2|y_1)p(y_1)} p(x).$$

Pokud  $Y_1, Y_2$  interpretujeme jako výsledky dvou nezávislých experimentů  $p(y_1, y_2|x) = p(y_1|x)p(y_2|x)$ , můžeme výraz přepsat na

$$p(x|y_1, y_2) \propto p(y_2|x)p(x|y_1).$$

Vztah můžeme ještě zobecnit i pro vektor n.v.  $Y_{1:n}$ :

$$p(x|y_{1:n}) \propto p(y_n|x)p(x|y_{1:n-1}).$$



Vzorec naznačuje, že zpracování většího počtu dat můžeme rozložit do více kroků, kdy v každém kroku využijeme aposteriorní hustotu z kroku předchozího jako hustotu apriorní.

Tohoto bychom mohli využít např. v situaci, kdy zpracováváme rozsáhlá data. Díky rozložení do kroků můžeme v některých případech snížit paměťovou náročnost jednotlivých výpočtů. Také si povšimneme, že nevyžadujeme předpoklad homogenity dat ( $Y_k$  a jejich rozdělení mohou být obecně různé divoké objekty), stačí nám znát model  $p(y_k|x); \forall k$ . Nehomogenita je totiž další problém, který souvisí s analýzou tzv. velkých dat [‘big data’].

Další výhodu nám sekvencializace přináší v situaci, kdy nám data postupně přibývají v čase. S každými novými daty  $y_k$  chceme najít aposteriorní hustotu  $p(x|y_{1:k})$ .

Ať už je náš případ jakýkoliv můžeme úlohu sekvenčního zpracování dat formulovat následovně:

**Úloha 3.1.** (sekvenční zpracování dat)

Buď  $(Y_k)_{k \in \mathbb{N}}$  posloupnost veličin  $Y_k$  rozdělených podle  $p(y_k|x)$ , tj. podmíněně nezávislých na  $Y_{1:k-1}$  za podmínky  $x$ .

Buď  $X_0 \sim p(x)$  apriorní hustota.

Pro všechna  $k \in \mathbb{N}$  chceme aproximovat rozdělení  $p(x|y_{1:k}) \propto p(x) \prod_{i=1}^k p(y_i|x)$ .

Pro obecný model ovšem neexistuje řešení bayesovské indukce v uzavřeném tvaru. Ve speciálním případě, kdy k modelu existuje konjugovaný systém a apriorní rozdělení do něj patří, můžeme sekvenční úlohu řešit postupnou úpravou parametrů tohoto systému. V opačném případě využijeme k řešení úlohy numerické metody.

### 3.2 Skryté Markovské modely

Zatím jsme v úloze bayesovské indukce nepředpokládali nic o skryté veličině  $X$ . V této podkapitole uvedeme úlohy předpokládající speciální tvar n.v.  $X$ . Zaměříme se na případy, kdy je  $X$  náhodný vektor. V obecnosti výše uvedené úvahy stále projdou, ale stejně jako vůči datům i zde se můžeme pokusit redukovat výpočetní náročnost odhadu za pomoci struktury modelu.

Představme si situaci, kdy data generujeme měřením polohy pohybujícího se předmětu v různých časech. Náhodné veličiny reprezentující tato data zřejmě nezávislé nebudou. Náhodným vektorem  $X$  můžeme modelovat polohou předmětu v časech měření. Označme n.v.  $X_k$  polohu předmětu v čase  $k$ -tého měření a n.v.  $Y_k$  naměřenou hodnotu v čase  $k$ . Speciálně ještě zavedme n.v.  $X_0$  reprezentující polohu předmětu v čase 0 (tj. před začátkem měření).

(Nutno zmínit, že poloha pohybujícího se předmětu je jen jednou konkrétní interpretací stochastického procesu a v této části slouží jen pro intuitivnější představu.)

Pro většinu praktických aplikací můžeme vyslovit další předpoklad o podmíněné nezávislosti pozorování  $Y_k$  na  $Y_{1:k-1}$  a  $X_{0:k-1}$  za podmínky  $X_k = x_k$ . Rozdělení souboru pozorování pak můžeme faktorizovat na

$$p(y_{1:k}|x_{0:k}) = \prod_{n=1}^k p(y_n|x_n).$$

Pro řešení úlohy bayesovské indukce na  $X_{0:k}$  potřebujeme ještě znát apriorní rozdělení  $p(x_{0:k})$ .

Vektory náhodných veličin  $X_k$ , které reprezentují nějakou formu vývoje stavu v reálném světě, často modelujeme jako tzv. Markovské řetězce (někdy označované procesy s konečnou pamětí). Markovským řetězcem dále budeme rozumět Markovský proces 1. řádu, ve kterém pro všechna  $n \in \mathbb{N}$  platí:

$$p(x_n|x_{0:n-1}) = p(x_n|x_{n-1}).$$

Chceme tím vyjádřit, že budoucí vývoj systému bude závislý pouze na současném stavu (obdobně jako v případě Cauchyho úlohy). Výhodou tohoto modelu je možná faktorizace sdružené pravděpodobnosti procesu

$$X_{0:k} \sim p(x_{0:k}) = p(x_0) \prod_{n=1}^k p(x_n|x_{n-1}). \quad (6)$$

Model vývoje systému nám poskytne apriorní rozdělení pro bayesovskou indukci.

Model, který jsme právě popsali, se v oblasti stochastického modelování vyskytuje častěji. Nazývá se skrytý Markovský model ([‘Hidden Markov model’] - HMM). Většina algoritmů, které budeme později popisovat vzniká právě aplikací obecnějších principů na tento model.

**Definice 3.2.** (Skrytý Markovský model)

Buď  $(X_k)_{k \in \mathbb{N}_0}$  Markovský řetězec.

Buď  $(Y_k)_{k \in \mathbb{N}}$  posloupnost veličin  $Y_k$  podmíněně nezávislých na  $Y_{1:k-1}, X_{0:k-1}$  za podmínky  $X_k = x_k$ .  
Proces  $X_0 \cup (X_k, Y_k)_{k \in \mathbb{N}}$ , jehož sdružené rozdělení můžeme pro všechna  $n \in \mathbb{N}$  zapsat ve tvaru:

$$p(x_{0:n}, y_{1:n}) = p(x_0) \left( \prod_{k=1}^n p(x_k|x_{k-1}) \right) \left( \prod_{k=1}^n p(y_k|x_k) \right) \quad (7)$$

nazveme skrytý Markovský model (HMM).

HMM definuje soustava hustot

$$\begin{cases} p(x_0) & ; \text{ (apriorní hustota počátečního stavu)} \\ p(x_k|x_{k-1}) & ; \text{ (přechodová pravděpodobnost skrytého řetězce)} \\ p(y_k|x_k) & ; \text{ (model pozorování)} \end{cases} \quad (8)$$

Libovolná aposteriorní rozdělení skrytých veličin (míněno neznámých) v bayesovské indukci nyní můžeme vyjádřit jako podmíněná rozdělení v HMM.

HMM nalézá uplatnění při modelování mnoha nestacionárních úloh. Soubor různých aplikací nalezne čtenář např. v [7, 8].

### 3.2.1 Optimální filtrace

Standardní situací vedoucí k aplikaci HMM je úloha sledování. Je-li skrytou veličinou stav nějakého evolučního systému, můžeme se ptát v jakém je stavu v okamžiku pozorování (jelikož měření je obecně zatíženo šumem daným modelem  $p(y_k|x_k)$ ). V sekvenčním případě úlohu interpretujeme jako sledování stavu.

Aplikace jsou zřejmé v oblasti automatizace řízení, kde se řídicí počítač musí rozhodovat v podmínkách neurčitosti. Aby automat provedl nějakou akci (např. spuštění požárního systému) potřebuje vědět, jestli došlo k podmínkám, které akci spouští (např. stav = hoří). V tomto případě bychom n.v.  $Y_k$  interpretovali jako sensorické vstupy (např. data z kouřových detektorů). Obecně můžeme vyslovit předpoklad, že každý automatický systém, který aktivně interaguje z naším fyzickým světem, bude

pro rozhodování vyžadovat informace o současném stavu systému - reality. Příklady praktických aplikací jsou např. lokalizace robota, který se má samostatně přesouvat v netriviálních oblastech (např. mezi místnostmi), klasifikace objektů na videovém záznamu nebo odhad polohy letadel z radarových dat.

Úloha sledování se nazývá také úloha optimální filtrace [‘optimal filtering’].

**Úloha 3.3.** (Optimální filtrace)

Mějme zadán HMM pro náhodné veličiny  $(X_k)_{k \in \mathbb{N}_0}; (Y_k)_{k \in \mathbb{N}}$ .

Pro všechna  $k \in \mathbb{N}$  chceme aproximovat rozdělení n.v.  $X_k | Y_{1:k} \sim p(x_k | y_{1:k})$ .

Ve speciálních případech lze opět sestavit konjugované systémy a úlohu 3.3 řešit sekvenční úpravou postačujících statistik. Významnou třídu algoritmů, které problém řeší pro lineární gaussové modely, jsou tzv. Kalmanovy filtry. Za těchto dodatečných předpokladů je aposteriorní rozdělení normální a v každém kroku nám stačí přepočítat střední hodnoty a kovarianční matici. Lineární gaussové modely jsou dostatečně obecné pro modelování spousty problémů. Proto stručně připomínáme zlatou větu o Monte Carlo algoritmech - pokud je nemusíte použít, nedělejte to. Více o Kalmanových filtrech lze nalézt v [9, 20].

My se budeme zabývat obecnější situací, která povede na aplikaci MC algoritmů. Obecnější modely se často označují jako nelineární negaussové modely. Existuje více možností, jak úlohu generování vzorků formulovat a řešit a více o nich bude řeč v kapitole 4. Numerické metody se opírají následující rekurní vztahy v HMM:

$$p(x_n | y_{1:n}) \propto p(y_n | x_n) p(x_n | y_{1:n-1}), \quad (9)$$

kde  $p(x_n | y_{1:n-1}) = \int p(x_n | x_{n-1}) p(x_{n-1} | y_{1:n-1}) dx_{n-1}$  je hustota predikce a  $p(y_n | x_n)$  model pozorování. Algoritmus založený na této rekurzi sekvenčně provádí predikci (v IS návrhové rozdělení) a tu upravuje pomocí získaných dat (v IS přiřazení vah).

Druhou možností je využití snadného výpočtu marginál z empirických rozdělení a úlohu filtrace převést na vzorkování z rozšířených prostorů - celé trajektorie až do času měření. Využívá této rekurze:

$$p(x_{0:n} | y_{1:n}) \propto p(y_n | x_n) p(x_n | x_{n-1}) p(x_{0:n-1} | y_{1:n-1}). \quad (10)$$

Základem této metody je postupné vzorkování ze vnořených prostorů. Nové vzorky vytvoříme tak, že protahujeme trajektorie vzorků z předchozího kroku. Když si to rozebereme, tak princip je podobný předchozímu postupu. Umožní nám ale v průběhu algoritmu pracovat i s historií trajektorie vzorku a využít několik postupů pro zlepšení konvergenčních vlastností.

### 3.2.2 Optimální vyhlazování

V případě, kdy nás zajímá poloha v nějakém čase za předpokladu, že máme k dispozici data pozorování až do nějakého pozdějšího času bude situace složitější, jelikož rozdělení skrytého procesu v daném čase je z modelu závislé i na následujícím stavu. Takovouto úlohu můžeme interpretovat jako aproximaci rozdělení celé trajektorie.

Budeme se snažit nalézt efektivní postupy, které nám umožní snížit náročnost výpočtu využitím struktury HMM.

**Úloha 3.4.** (optimal smoothing)

Mějme zadán HMM na náhodných veličinách  $(X_k)_{k \in \mathbb{N}_0}; (Y_k)_{k \in \mathbb{N}}$ .

Pro pevné  $T \in \mathbb{N}$  a libovolné  $k \in [0, T]$  chceme aproximovat rozdělení n.v.  $X_k|Y_{1:T} \sim p(x_k|y_{1:T})$ .

Ač můžeme řešení opět získat marginalizací  $p(x_k|y_{1:T}) = \int p(x_{0:T}|y_{1:T})dx_{-k}$ , v případě empirické distribuce  $\hat{\pi}_T^N(dx)$  získané řešením úlohy filtrace bude takto vzniklá aproximace reprezentována jen malým počtem odlišných vzorků. Za toto může mezikrok algoritmu (převzorkování), který musíme ve většině případů uvést pro zaručení konvergenčních vlastností. Jinými slovy takto získáme špatnou aproximaci hledaného rozdělení. Pro jiné metody generování vzorků z rozšířených prostor (např. MCMC) tyto problémy nastat nemusejí, přesto se pokusíme i tuto úlohu řešit pomocí sekvenčních algoritmů.

### 3.2.3 General state-space model

Připomeňme zmíněnou náročnost odhadu statického parametru. General state-space model (GSSM) je zobecněním HMM. Rozdílem je uvedení další n.v.  $\theta$ , která reprezentuje neznámý statický parametr.

Málokdy se totiž stává, že přesně známe modely pohybu a pozorování. Řešením je uvažovat třídu modelů závislých na neznámém parametru.

$$\begin{cases} p(\theta) \\ p(x_0|\theta) \\ p(x_k|x_{k-1}, \theta) \\ p(y_k|x_k, \theta) \end{cases}$$

Aposteriorní rozdělení parametru  $\theta$  se tedy také stává předmětem odhadu, což nám ovšem úlohu významně komplikuje.

Pro tento model obecně nelze využít efektivní algoritmy jako v případě problémů (3.3) a (3.4). Jednou možností, jak úlohu řešit, je aplikace MCMC algoritmů na sdružené rozdělení  $p(\theta, x_{1:m}|y_{1:m})$ . Existují i sekvenční algoritmy, které úlohu řeší např. sestavením aproximace modelu převedením  $\theta$  na proces (s  $\theta|Y_{1:k}$  pracujeme jako s procesem  $\theta_k|Y_{1:k}$ ). I zde můžeme sestavit algoritmy využívající postačujících statistik  $\theta|Y_{1:k}$ .

Model GSSM zmiňujeme pro úplnost, jelikož je v současné době předmětem studií MC teoretiků. Více o problému i metodách řešení lze nalézt např. v [3, 7].

## 4 Sekvenční Monte Carlo algoritmy

Sekvenční Monte Carlo algoritmy jsou širokou třídou metod pro tvorbu vzorků z obecných sekvencí rozdělání. Historicky se vyvinuly z potřeby řešit úlohu optimální filtrace (úloha 3.3).

Podtřídou SMC algoritmů jsou částicové filtry [‘particle filter’] (PF). Jejich zavedení není jednoznačné. Většinou se setkáme s jejich zavedením pouze pro úlohu optimální filtrace. Někde jsou zase zaměňovány za SMC, které zde budeme chápat obecněji.

Nejprve uvedeme PF obdobně jejich zavedení v [8] (kde je mj. uvedena ucelená teorie aplikace PF pro řešení optimální filtrace a vyhlazování na HMM). Následně algoritmus zobecníme a formulujeme SMC algoritmus, který v popisu zahrne i jiné struktury modelu (vč. úlohy sekvenčního odhadu statického parametru). Dále pak ukážeme, jak lze z obecných vztahů opět získat algoritmus pro řešení úlohy optimální filtrace.

V závěru kapitoly zmíníme několik úprav, které mají za cíl zlepšit konvergenční vlastnosti algoritmů.

### 4.1 Particle filtry pro optimální filtraci

Rozvoj sekvenčních metod nastal při řešení úlohy optimální filtrace (úloha 3.3). Připomeňme, že chceme aproximovat sekvenci rozdělání  $\{\pi_k(x_k) = p(x_k|y_{1:k})\}_{k \in \mathbb{N}_0}$ .

#### 4.1.1 Sekvenční importance sampling

Sekvenční metody, jimiž se budeme zabývat, využívají k tvorbě aproximací metodu IS. V každém kroku hledáme soubor vážených vzorků, abychom aproximovali  $\pi_k(x_k)$  a odhadli

$$I_{\pi_k}(\varphi) = \mathbb{E}_{\pi_k}(\varphi(X_k)) \approx \mathcal{E}_{\pi_k}^{IS}(\varphi(X_k)) = \sum_{i=1}^N W_k^i \varphi(X_k^i).$$

Připomeňme IS aproximaci rozdělání (2.6). V každém kroce musíme vytvořit vzorky z voleného návrhového rozdělání  $q_k(dx_k)$  a napočítat váhy  $W_k^i \propto w_k(X_k^i) = \frac{\pi_k(x_k)}{q_k(x_k)}$ . Váhová funkce je jednoznačně daná volbou návrhového rozdělání a proto máme jedinou volnost ve výběru návrhové hustoty.

Pro získání nápadu, jak volit návrhové hustoty, využijeme vztahu pro marginální rozdělání (speciálně pro HMM ve tvaru):

$$p(x_k|y_{1:k}) = \int p(x_{0:k}|y_{1:k}) dx_{-k},$$

kde  $p(x_{0:k}|y_{1:k})$  je aposteriorní rozdělání celé trajektorie (7). Hledaná hustota vznikne jeho marginalizací. O marginalizacích IS aproximací hovoří následující lemma:

**Lemma 4.1.** (*marginály MC aproximací*)

*Bud’  $\{W^i, X_{1:k}^i\} \sim \pi(x_{1:k})$  soubor vážených vzorků.*

*Pak marginální rozdělání  $\pi(x_k)$  můžeme aproximovat “vynecháním ostatních proměnných”*

$$\hat{\pi}^N(dx_k) = \sum_{i=1}^N W^i \delta_{X_k^i}(dx_k)$$

Původní úlohu aproximace  $\{\pi_k(x_k)\}_{k \in \mathbb{N}_0}$  jsme tedy převedli na úlohu aproximace  $\{p(x_{0:k}|y_{1:k})\}_{k \in \mathbb{N}_0}$  a její marginalizaci. Zbývá nám vybrat návrhové rozdělení pro tvorbu vzorků.

Předpokládáme, že umíme vytvořit soubor vážených vzorků  $\{W_0^i, X_0^i\} \sim p(x_0)$  (počátečního rozdělení HMM). Dále postupujeme induktivně. V každém kroku předpokládáme, že máme k dispozici soubor vážených vzorků z minulého kroku.

Particle filtry jsou specifické použitím tzv. protahování trajektorií. Můžeme na to pohlížet jako na výběr návrhové hustoty tvaru  $q_k(x_{0:k}) = q_k(x_k|x_{0:k-1})q_{k-1}(x_{0:k-1})$ . V každém kroku vytvoříme pro všechna  $i$  vzorky  $X_{0:k}^i$  z tohoto rozdělení tak, že  $X_{0:k-1}^i$  opíšeme ze vzorku  $X_{0:k-1}^{i-1}$  z minulého kroku a  $X_k^i$  vygenerujeme z rozdělení  $q_k(x_k|X_{0:k-1}^i)$ . Vzniklý sdružený vzorek bude rozdělený podle  $X_{0:k}^i \sim q_k(x_{0:k})$  a zbývá k němu dopočítat váhu.

Optimálním návrhovým rozdělením protahovaného vzorku pro filtraci na HMM - ve smyslu minimalizace vah vzorků - je  $q_k^{opt}(x_k|X_{0:k-1}^i) = p(x_k|X_{k-1}^i, y_k)$  [8]- rozdělení, které bere v úvahu i pozorování v čase  $k$ .

Pro váhovou funkci platí (díky rekurzi (10)):

$$\begin{aligned} w_k(x_{0:k}) &\propto \frac{p(x_{0:k}|y_{1:k})}{q_k(x_{0:k})} = \frac{p(y_k|x_k)p(x_k|x_{k-1})}{q_k(x_k|x_{0:k-1})} \frac{p(x_{0:k-1}|y_{1:k-1})}{q_{k-1}(x_{0:k-1})} \\ &= \frac{p(y_k|x_k)p(x_k|x_{k-1})}{q_k(x_k|x_{0:k-1})} w_{k-1}(x_{0:k-1}) \end{aligned}$$

Jelikož jsme pro všechny vzorky část trajektorie  $X_{0:k-1}^i$  opisovali a předpokládali, že vzorky z minulého kroku jsou řádně vážené (def. 2.10), bude druhá část úměrná váze vzorku v předcházejícím kroku:  $w_{k-1}(X_{0:k-1}^i) \propto W_{k-1}^i$ .

Vztah pro úpravu vah takto generovaných (protahování) vzorků můžeme kompaktně zapsat:

$$W_k^i \propto \alpha_k(X_{0:k}^i) W_{k-1}^i,$$

kde  $\alpha_k(x_{0:k}) = \frac{p(y_k|x_k)p(x_k|x_{k-1})}{q_k(x_k|x_{0:k-1})}$  nazveme inkrementální váha.

Poznámka: často se tvar inkrementální váhy zapisuje ve formě  $\alpha_k(x_{0:k}) = \frac{\pi_k(x_{0:k})}{\pi_{k-1}(x_{0:k-1})q_k(x_k|x_{0:k-1})}$ . V takovém výrazu se ovšem vyskytuje ve jmenovateli  $\pi_{k-1}(x_{0:k-1})$  a při numerickém řešení se může stát, že  $\pi_{k-1}(x_{0:k-1}) = 0$ . Toto musíme v algoritmu ošetřit (např. brzy uvedeme krok převzorkování, který je schopen takové částice odstranit).

Algoritmus Sekvenční importance sampling (SIS) můžeme shrnout:

Hledanou aproximaci získáme marginalizací ve tvaru

$$\hat{\pi}_k^N(dx_k) = \sum_{i=1}^N W_k^i \delta_{X_k^i}(dx_k).$$

#### 4.1.2 Degenerace vzorků a převzorkování

Oproti algoritmu IS je v sekvenčních metodách obtížnější zaručit konvergenci.

**Algoritmus 4** Sekvenční importance sampling**pro**  $k = 1$ **Vytvoř** soubor vážených vzorků  $\{W_0^i, X_0^i\} \sim p(x_0)$ **pro**  $k > 1; \forall i = 1 : N$ **Vytvoř** vzorek  $X_k^i \sim q_k(x_k | X_{0:k-1}^i)$  a  $X_{0:k}^i \leftarrow [X_{0:k-1}^i, X_k^i]$ **Vypočti** váhy  $W_k^i \propto W_{k-1}^i \alpha_k(X_{0:k}^i) = W_{k-1}^i \frac{p(y_k | x_k) p(x_k | x_{0:k-1})}{q_k(x_k | x_{0:k-1})}$ **Výstupem** kroku  $k$  je aproximace  $\hat{p}^N(x_{0:k} | y_{1:k})$ .**Lemma 4.2.** (o rozptylu vah)*Pro sdružené hustoty  $p(a, b), q(a, b)$ ,  $p \ll q$  platí ([7, 15]):*

$$D_{q(a,b)}\left(\frac{p(A,B)}{q(A,B)}\right) \geq D_{q(a)}\left(\frac{p(A)}{q(A)}\right),$$

*kde  $p(a)$  a  $q(a)$  představují marginály  $\int p(a, b)db$ ,  $\int q(a, b)db$  resp..*

V algoritmu 4 se kvůli způsobu konstrukce jedná o nerovnost

$$D_{q_k}[w_k(X_{0:k})] = D_{q_k}[\alpha_k(X_{0:k})w_{k-1}(X_{0:k-1})] \geq D_{q_{k-1}}[w_{k-1}(X_{0:k-1})].$$

Výpočtem vah ze sdružených vzorků způsobíme navýšení rozptylu vah, na nichž závisí rozptyl výsledných odhadů. Zpravidla platí, že roste exponenciálně. Proto je třeba představit další důležitý prvek sekvenčních metod - převzorkování [‘resampling’].

Představme si, že nám v nějakém kroku velmi naroste rozptyl vah. Tyto vzorky chceme v dalším kroce využít pro vytvoření nových vzorků, ale podle lemma 4.2 rozptyl vah takto vytvořených vzorků ještě vzroste. Zdá se výhodné najít způsob, jak před samotným vytvářením nových vzorků tuto dolní hranici snížit. Výběr optimálního návrhového rozdělení  $q^*(x_{0:k}) = p(x_{0:k} | y_{1:k})$  vede na váhovou funkci  $w(x_{0:k}) = 1$  a  $W_k^i = \frac{1}{N}$  (nulový rozptyl). Pokusíme se o něco ve smyslu vnořeného IS algoritmu, který vytváří vzorky s pomocí empirického návrhového rozdělení  $q(dx_{0:k}) = \hat{\pi}^N(dx_{0:k})$ . Takto můžeme interpretovat náhodný výběr s vrácením z vektoru  $X_{0:k}^{1:N}$  i.i.d. podle  $q_k(x_{0:k})$ , kde vzorek  $X_{0:k}^i$  vložíme do nového výběru s pravděpodobností  $W_k^i$ .

**Definice 4.3.** (převzorkování)Buď  $\{W^i, X^i\} \sim \pi(x); X^i \sim q(x)$  soubor vážených vzorků.Provedeme-li  $M$  výběrů s vrácením  $X'$  z vektoru  $X^{1:N}$ , kde prvek  $i$  vybereme s pravděpodobností  $W^i$ , získáme soubor vzorků  $X'^{1:M}$  i.i.d. podle  $\pi(x)$  (tedy  $W^i = \frac{1}{M}; \forall i$ ).

Tento proces nazveme převzorkování.

Převzorkování sníží rozptyl vah na nulu. Mezi algoritmy, které jej provádějí patří např. Multinomial resampling, Systematic resampling nebo Residual resampling [6, 8].

Převzorkování nás s vysokou pravděpodobností zbaví vzorků s nízkými váhami (vzorků s vahou 0 nás zbaví jistě za předpokladu, že soubor obsahuje alespoň jeden další vzorek s nenulovou vahou).

Převzorkování ovšem není samospásné. Také má negativní vliv na rozptyl výsledného odhadu. Přesto se zdá být jeho použití nutnou podmínkou pro zaručení konvergence sekvenčních algoritmů.

Převzorkování je také důvod, proč nemůžeme řešení úlohy optimal smoothing (úloha 3.4) získat marginalizací aproximace řešení spojené úlohy optimální filtrace -  $\hat{p}^N(dx_{0:T}|y_{1:T})$ . Při filtraci většinou nevytváříme nové vzorky pro celou historii procesu. Při tvorbě nových vzorků se historie opisuje a proto po násobném zahazování vzorků s nízkými vahami způsobeném převzorkováním budou rozdělení  $X_k$  v dřívějších časech aproximována pomocí jen několika málo odlišných vzorků, 1 není výjimkou. Což má ovšem špatný vliv na konvergenci MC odhadu (konvergenci pro  $N \rightarrow \infty$ ).

Ač je užitečným krokem, v algoritmech se snažíme převzorkovávat co nejméně. Kritériem pro převzorkování bývá míry efektivity vzorků Effective Sample Size [7] ( $ESS = \left(\sum_{i=1}^N (W^i)^2\right)^{-1}$ ).  $ESS$  nabývá hodnot mezi  $1..N$ . Pokud v průběhu výpočtu klesne pod předem stanovenou mez (obvykle  $\frac{N}{2}$ ), převzorkujeme. Pravidlem také bývá provádět v obecném kroku resampling před generováním nových vzorků pomocí protahování (proto jej provádíme).

V souvislosti s odhadem funkcionálu (1) navíc platí, že nižšího výsledného rozptylu dosáhneme, pokud k odhadu využijeme vzorky, které převzorkované nejsou (převzorkování rozptyl zvýší) [8].

#### 4.1.3 Sekvenční Monte Carlo pro optimální filtraci

Výše zmíněné úvahy můžeme shrnout do obecného algoritmu, který je v [8] pojmenován SMC s adaptivním převzorkováním.

---

##### Algoritmus 5 Sekvenční Monte Carlo s adaptivním převzorkováním

---

**Pro**  $k = 1$

**Vygeneruj** vzorky  $X_0^i \sim q_0(x_0)$  a spočti váhy  $W_0^i \propto w(X_0^i)$

**Pro**  $k > 1$ ;

**Je-li**  $ESS < \frac{N}{2}$ ; Převzorkuj  $X_{0:k-1}^{1:N}$  na soubor vzorků i.i.d. podle  $p(x_{0:k-1})$

$\forall i = 1 : N$

**Vytvoř** vzorek  $X_k^i \sim q_k(x_k|X_{0:k-1}^i)$  a  $X_{0:k}^i \leftarrow [X_{0:k-1}^i, X_k^i]$

**Vypočti** váhy  $W_k^i \propto W_{k-1}^i \alpha_k(X_{0:k}^i) = W_{k-1}^i \frac{p(y_k|x_k)p(x_k|x_{k-1})}{q_k(x_k|x_{0:k-1})}$

**Výstupem** kroku  $k$  je aproximace  $\hat{p}^N(dx_{0:k})$  (soubor vážených vzorků).

---

Analýza vlivu převzorkování na rozptyl výsledného odhadu (1) je dostupná pro HMM v [8].

## 4.2 Obecný sekvenční Monte Carlo algoritmus

V předchozí podkapitole jsme předvedli algoritmus pro sekvenční řešení úlohy optimální filtrace. V této podkapitole se pokusíme sekvenční metody zobecnit na širší škálu úloh a ukážeme, jak z tohoto zobecnění opět získat algoritmy PF.



Budeme se zabývat úlohou tvorby vzorků ze sekvenční rozdělení  $\{\pi_k\}_{k \in \mathbb{N}_0}$ .

Předpoklad 1: Každá míra má hustotu  $\frac{d\pi_k}{ds_k} = \pi_k(s_k)$ .

Předpoklad 2: Každou hustotu  $\pi_k(s_k)$  můžeme zapsat ve tvaru  $\pi_k(s_k) = \frac{\gamma_k(s_k)}{Z_k}$ , kde  $\gamma_k(s_k)$  je nevlastní hustota na  $R(S_k)$ , kterou umíme bodově vyčíslit a  $Z_k \in \mathbb{R}$  má význam normalizační konstanty (může být neznámá).

**Induktivní přístup** Celý algoritmus iteračně opakuje stejný postup, jen v každém kroku mění rozdělení se kterými pracuje.

1. V prvním kroku vytvoříme soubor vážených vzorků  $\{W^i, S_0^i\} \sim \pi_0(s_0)$  (k tomu využijeme např. IS [MCMC ne! ta tvoří závislé vzorky]).
2. Poté v každém kroku využijeme totožný postup, kdy pomocí vzorků získaných v předchozím kroku  $\{W_{k-1}^i, S_{k-1}^i\} \sim \pi_{k-1}(s_{k-1})$  vytvoříme nové vzorky  $\{W_k^i, S_k^i\} \sim \pi_k(s_k)$ .

Všechny metody a principy proto budeme popisovat pro obecný krok  $k$ . Při implementaci pak stačí aplikovat popsání principy postupně pro všechna  $k$ .

#### 4.2.1 Transformace rozdělení

Dále se pokusíme popsat SMC jako posloupnost transformací vzorků. Celý iterační algoritmus pak můžeme vágně chápat jako proces, který posouvá vzorky z cílového rozdělení minulého kroku do cílového rozdělení současného kroku - transformuje jeden soubor ve druhý.

K popisu transformací v prostorech cílových rozdělení využijeme přechodová jádra. Zmínili jsme je už v souvislosti s MCMC, nyní ale budeme potřebovat s pojmem nakládat exaktněji. Formálně se jedná pouze o přeznačení podmíněného rozdělení.

**Definice 4.4.** (přechodové jádro)

Bud'  $X, X'$  náhodné veličiny.

Pokud lze sdruženou hustotu  $p(x, x')$  zapsat ve tvaru  $p(x, x') = p(x)t(x'|x)$ , kde  $t(x'|x)$  je podmíněné rozdělení  $X'$ , pak

$$T(x, x') := t(x'|x)$$

nazveme přechodovým jádrem.

$T(x, dx')$  je pro všechna pevná  $x$  rozdělení pravděpodobnosti na  $\mathcal{A}(X')$ .

Pro zkrácený zápis transformací zavedeme zápis:

Označme pro hustotu  $p(x)$  a přechodové jádro  $T(x, x')$  transformované rozdělení  $(pT)$ :

$$(pT)(dx') = \int T(x, dx')p(dx).$$

#### 4.2.2 Vytváření částic

V každém kroku chceme vytvářet vážené vzorky. O IS už víme, že rozptyl výsledného odhadu  $D_q[\mathcal{E}_\pi^{IS}(\varphi)]$  je pro obecnou funkci  $\varphi \in C_b(R(S_k))$ , kde  $C_b$  značí prostor spojitých a omezených

funkcí, úměrný rozptylu váhové funkce  $w = \frac{d\pi}{dq}$ . Lidsky řečeno, čím je návrhová hustota podobnější cílové hustotě, tím kvalitnější odhady dostaneme. Optimální volba návrhové hustoty je  $q^*(s_k) = \pi_k(s_k)$ . Z takového rozdělení ovšem neumíme generovat vzorky. Snažíme se proto nalézt suboptimální návrhovou hustotu - aproximaci  $\pi_k(s_k)$ .

Předpokládejme tedy že v obecném kroku  $k$  máme k dispozici soubor vážených vzorků  $\{W_{k-1}^i, S_{k-1}^i\} \sim \pi_{k-1}(s_{k-1})$ ;  $S_{k-1}^i \sim q_{k-1}(s_{k-1})$ .

Do algoritmu formálně vložíme mezikrok vytvoření sdružených vzorků. Zvolíme  $Q_k(s_{k-1}, s_k)$ , které nazveme návrhovým přechodovým jádrem. Sdružené vzorky budeme vytvářet tak, že ke každému vzorku  $S_{k-1}^i$  vytvoříme vzorek  $S_k^i \sim Q_k(S_{k-1}^i, s_k)$ . Řekneme, že vzorky propagujeme podle přechodového jádra  $Q_k(s_{k-1}, s_k)$ .

Takto vzniklý sdružený vzorek bude rozdělený:

$$[S_{k-1}^i, S_k^i] \sim q_k(s_{k-1}, s_k) := q_{k-1}(s_{k-1})Q_k(s_{k-1}, s_k).$$

Kdybychom teď chtěli počítat váhy jako Radonovy-Nikodýmovy derivace, rozdělení vzniklých vzorků  $S_k^i$  získáme marginalizací rozdělení sdružených vzorků (připomeňme značení  $q(b|a) = Q(a, b)$ )

$$q_k(s_k) = (q_{k-1}Q_k)(s_k) = \int q_k(s_k|s_{k-1})q_{k-1}(s_{k-1})ds_{k-1} \quad (11)$$

Pro vytvoření IS aproximace musíme napočítat váhu

$$W_k^i \propto \frac{\pi_k(S_k^i)}{q_k(S_k^i)} = \frac{\pi_k(S_k^i)}{(q_{k-1}Q_k)(S_k^i)} \quad (12)$$

Pro výpočet vah ovšem potřebujeme umět návrhovou hustotu vyčíslit v každém bodě  $s_k$ . I kdybychom se rozhodli vypočíst integrál numericky, tak jelikož nové vzorky vznikají rekurzivně, museli bychom v každém výpočtu vypočíst

$$q_k(s_k) = (q_{k-1}Q_k)(s_k) = (q_0Q_1Q_2\dots Q_k)(s_k).$$

Museli bychom tedy vyčíslovat alespoň  $k$ -rozměrný integrál. Jednou možností, jak problém řešit je využít dostupné MC aproximace a počítat odhady:

$$\begin{aligned} q_k(s_k) &= \mathbb{E}_{q_{k-1}}(Q_k(S_{k-1}, s_k)) \\ &\approx \mathcal{E}_{q_{k-1}}^{MC}(Q_k(S_{k-1}, s_k)) = \frac{1}{N} \sum_{i=1}^N Q_k(S_{k-1}^i, s_k) \end{aligned} \quad (13)$$

Krom chyby v odhadu navíc platí, že náročnost takového generování vzorků by byla  $\mathcal{O}(N^2)$ .

Uveďme pro úplnost speciální volbu přechodového jádra  $Q_k(s_{k-1}, s_k) = q_k(s_k)$  (nové vzorky ovšem stále mohou záviset na předchozích přes odhady parametrů rozdělení  $q_k(s_k)$ ). V literatuře tato jádra označují jako nezávislé návrhy [‘independent proposal’], jelikož nezávislejší přímo na vzorku, který protahujeme, a umíme je proto bodově vyčíslit. Taková volba je ale, převážně ve vysokorozměrných prostorech, velmi omezující.

Mezi další užívaná jádra patří např. jádro náhodné procházky, kde ale není zcela triviální určit optimální parametry procházky. Pokud je to možné, snažíme se při konstrukci návrhového rozdělení využít struktury cílové hustoty  $\pi_k(s_k)$ . V případech, kdy mají veličiny stejné obory hodnot můžeme využít jádro se stabilním rozdělením  $\pi_k = \pi_k Q_k$ , které by nám mělo v jistém smyslu generovat návrhy blíže hledaného rozdělení  $\pi_k$ .

### 4.2.3 Pomocná sdružená rozdělení

Uvedeme metodu, jak zjednodušit vztahy pro výpočet nových vah. Proto ovšem budeme muset zobecnit pojem váhové funkce. Vzpomeňme, že v IS jsme vytvářeli vážené vzorky pomocí známé Radon-Nikodýmovy derivace  $w(x)$ . Ta ovšem není jedinou možnou volbou váhové funkce pro získání korektního odhadu funkcionálu (1).

Zde, jelikož využíváme návrhové rozdělení tvaru  $(qQ_k)(s_k)$ , podíváme se na sdruženou hustotu  $[S_{k-1}, S_k]$ .

**Lemma 4.5.** *(o šikvných marginálách)*

*Mějme zadáno rozdělení  $\pi(a)$ , které chceme aproximovat.*

*Bud'  $p(a, b)$  taková sdružená hustota pravděpodobnosti, že  $\int p(a, b)db = \pi(a)$ .*

*Pro funkcionál  $I_\pi(\varphi)$  a  $\varphi \in C_b(R(A))$  platí:*

$$\begin{aligned} \int \varphi(a)\pi(a)da &= \int \varphi(a) \left( \int p(a, b)db \right) da \\ &= \int \varphi(a)p(a, b)dadb = \int \tilde{\varphi}(a, b)p(a, b)dadb, \end{aligned}$$

*kde  $\tilde{\varphi}(a, b) = \varphi(a)$  je rozšířením funkce  $\varphi$  na  $R(A) \times R(B)$ .*

Hledaný funkcionál tedy můžeme pro vhodné pomocné sdružené hustoty aproximovat také z aproximace rozdělení na takto rozšířeném prostoru.

Pokud se nám podaří nalézt vhodnou pomocnou hustotu  $p(s_{k-1}, s_k)$  ( $\int p(s_{k-1}, s_k)ds_{k-1} = \pi_k(s_k)$ ), váhy  $W_k^i$  můžeme vypočíst jako váhy sdruženého vzorku  $w(s_{k-1}, s_k) = \frac{p(s_{k-1}, s_k)}{q_{k-1}(s_{k-1})Q_k(s_{k-1}, s_k)}$ .

Takto počítané váhy nám zaručí řádně vážený soubor [15] a s tím asymptotickou nestrannost odhadu. Rozptyl vah - kvalita estimátoru - se ovšem bude lišit v závislosti na zvolené pomocné funkci  $p(s_{k-1}, s_k)$ . Optimální váha - ve smyslu minimálního rozptylu - má tvar [5]

$$w^{opt}(s_{k-1}, s_k) = \frac{\pi_k(s_k)}{q_k(s_k)}.$$

Z lemma (4.2) opět získáme odhad rozptylu pro libovolnou volbu  $p(s_{k-1}, s_k)$  zdola omezený právě rozptylem optimálních vah

$$\left( D_{q_k}[w(S_{k-1}, S_k)] = D_{q_{k-1} \times Q_k} \left[ \frac{p(S_{k-1}, S_k)}{q_{k-1}(S_{k-1})Q_k(S_{k-1}, S_k)} \right] \geq D_{q_k} \left[ \frac{\pi_k(S_k)}{q_k(S_k)} \right] = D_{q_k}[w^{opt}(S_{k-1}, S_k)] \right).$$

V SMC častěji užíváme tvar pomocí váhy v předcházejícím kroku

$$w(s_{k-1}, s_k) = \alpha_k(s_{k-1}, s_k) w_{k-1}(s_{k-1}),$$

kde  $\alpha_k(s_{k-1}, s_k) = \frac{p(s_{k-1}, s_k)}{\pi_{k-1}(s_{k-1}) Q_k(s_{k-1}, s_k)}$  nazveme inkrementální vahou stejně jako u SIS (alg. 4).

#### 4.2.4 Odvození PF

V této části se pokusíme odvodit algoritmus PF pro optimální filtraci na HMM z výše uvedených obecných vztahů. Výhodou postupu je formulace obecnějšího algoritmu, než jen pro HMM. Uvedeme dva postupy, které pro filtraci na HMM vedou k téměř totožnému algoritmu.

**Formulace pro Markovské procesy** Nejprve se zabýváme speciálním případem, kdy aproximujeme sekvenci rozdělání n.v. tvořících Markovský proces s počátečním rozdělením  $\pi_0(s_0)$  a přechodovou hustotou  $\pi_k(s_k | s_{k-1}) \stackrel{\text{ozn.}}{=} T_k(s_{k-1}, s_k)$ .

(Pro HMM volíme  $S_k \sim p(x_k | y_{1:k})$  a proto z modelu HMM platí  $T_k(x_{k-1}, x_k) \propto p(y_k | x_k) p(x_k | x_{k-1})$ ). Předpokládejme, že máme z minulého kroku k dispozici soubor vážených vzorků  $\{W_{k-1}^i, S_{k-1}^i\} \sim \pi_{k-1}(s_{k-1}); S_{k-1}^i \sim q_{k-1}(s_{k-1}^i)$ . Do algoritmu vložíme mezikrok vzorkování

$$[S_{k-1}^i, S_k^i] \sim q_{k-1}(s_{k-1}) Q_k(s_{k-1}, s_k)$$

(opět ke každému  $S_{k-1}^i$  vytvoříme  $S_k^i \sim Q_k(S_{k-1}^i, s_k)$  propagací).

Sdružené rozdělení můžeme pomocí řetězového pravidla vyjádřit  $\pi(s_{0:m}) = \pi_0(s_0) \prod_{k=1}^m T_k(s_{k-1}, s_k)$ .

Je patrné, že pokud bychom nevyužili pomocného sdruženého rozdělení, museli bychom váhy nových vzorků ve formě Radonových-Nikodýmových derivací odhadovat pomocí vztahu:

$$w(s_k) = \frac{\pi_k(s_k)}{q_k(s_k)} = \frac{\int T_k(s_{k-1}, s_k) \pi_{k-1}(s_{k-1}) ds_{k-1}}{\int Q_k(s_{k-1}, s_k) q_{k-1}(s_{k-1}) ds_{k-1}}.$$

Pomůžeme si volbou vhodného pomocného sdruženého rozdělení

$$p(s_{k-1}, s_k) := T_k(s_{k-1}, s_k) \pi_{k-1}(s_{k-1}) = \pi(s_{k-1}, s_k),$$

pro které ze struktury modelu (pro HMM se jedná o rekurzi (9)) platí  $\int p(s_{k-1}, s_k) ds_{k-1} = \pi_k(s_k)$ .

Sdruženým vzorkům přiřadíme váhy:

$$w(s_{k-1}, s_k) = \frac{\pi(s_{k-1}, s_k)}{q(s_{k-1}, s_k)} = \frac{T_k(s_{k-1}, s_k)}{Q_k(s_{k-1}, s_k)} \frac{\pi_{k-1}(s_{k-1})}{q_{k-1}(s_{k-1})},$$

což je mnohem jednodušší vztah oproti výpočtu bez pomocného kroku, který vede na podíl integrálů.

Vážené vzorky příslušné  $\pi_k(s_k)$  nyní můžeme získat marginalizací IS aproximace  $\hat{p}^N(ds_{k-1}, ds_k)$  podle lemma 4.1.

Nalezený vztah opět zapíšeme ve tvaru úpravy váhy vzorku z předchozího kroku:

$$w(s_{k-1}, s_k) = \alpha_{k-1}(s_{k-1}, s_k) w_{k-1}(s_{k-1}),$$

kde  $\alpha_k(s_{k-1}, s_k) = \frac{T_k(s_{k-1}, s_k)}{Q_k(s_{k-1}, s_k)}$  je inkrementální váha (u HMM nabývá inkrementální váha tvaru  $\alpha_k(x_{k-1}, x_k) = \frac{p(y_k|x_k)p(x_k|x_{k-1})}{q_k(x_k|x_{k-1})}$ ) a  $w_{k-1}(s_{k-1})$  váhová funkce z minulého kroku.

Po marginalizaci získáme soubor vzorků  $S_k^i \sim q_k(s_k) = (q_{k-1}Q_k)(s_k)$  a váhy  $W_k^i \propto \alpha_k(S_{k-1}^i, S_k^i)W_{k-1}^i$ . Vzorky  $\{W_k^i, S_k^i\}$  budou díky konstrukci řádně vážené podle  $\pi_k(s_k)$  (a lze je využít v následujícím kroku metody).

Odvodili jsme tedy PF pro optimální filtraci. Získanou metodu lze využít v širším kontextu - pro vzorkování z Markovských procesů.

**Formulace pro proces se známým rozdělením podmíněným historií** Mějme náhodný proces  $(X_k)_{k \in \mathbb{N}_0}$  s přechodovou hustotou

$$\pi_k(x_k|x_{0:k-1}) \propto \gamma_k(x_k|x_{0:k-1})$$

(obdoba znalosti úplných podmíněných rozdělení u Gibbsova sampleru). Markovský proces je konkrétní volbou takového procesu ( $\pi_k(x_k|x_{0:k-1}) = \pi_k(x_k|x_{k-1})$ ).

Definujme pomocnou náhodnou veličinu  $S_k \sim \pi_k(x_{0:k}) := \pi_0(x_0) \prod_{n=1}^k \pi_n(x_n|x_{0:n-1})$ . Můžeme si všimnout, že  $R(S_{k-1})$  je vnořený do prostoru  $R(S_k)$ .

Označme  $S_{k,i}$  veličinu  $X_i \subset S_k$  (budeme chtít indexovat jednotlivá  $X_i|_{i \in [0,k]}$  v pomocné n.v.  $S_k$ )

Všimněme si, že díky konstrukci triviálně platí  $\pi_k(x_k) = \int \pi_k(x_{0:k}) dx_{-k}$ . Pro generování vzorků z cílového rozdělení můžeme zneužít vzorky z této pomocné n.v. pro vytvoření MC aproximace marginály podle lemma (4.1).

Dále můžeme postupovat podle obecného SMC, potřebujeme zvolit přechodové jádro a vypočítat váhy.

Speciální formou přechodového jádra reprezentujícího protahování částic je

$$Q_k(s_{k-1}, ds_k) = \delta_{s_{k-1}, 0:k-1}(ds_{k,0:k-1})q_k(ds_{k,k}|s_{k-1,0:k-1}).$$

Sdružené vzorky opět tvoříme tak, že ke každému  $S_{k-1}^i$  vytvoříme  $S_k^i \sim Q_k(S_{k-1}^i, s_k)$ . Sdružené návrhové rozdělení po sobě jdoucích pomocných n.v. lze zapsat ve tvaru:

$$q(ds_{k-1}, ds_k) = q_{k-1}(ds_{k-1})Q_k(s_{k-1}, ds_k).$$

Nyní musíme zvolit vhodnou pomocnou funkci  $p(s_{k-1}, s_k)$ . Pro korektnost postupu požadujeme  $p < q$  a proto můžeme volit

$$p(ds_{k-1}, ds_k) = \delta_{s_{k,0:k-1}}(ds_{k-1,0:k-1})\bar{\gamma}_{k-1}(s_{k-1,0:k-1})\bar{\gamma}_k(ds_{k,k}|s_{k-1,0:k-1}),$$

kde  $\bar{\gamma}_n(s_n) \propto \bar{\pi}_n(s_n) \forall n$ .

Můžeme ověřit, že

$$\int p(s_{k-1}, s_k) ds_{k-1} = \bar{\gamma}_{k-1}(s_{k,0:k-1})\gamma_k(s_{k,k}|s_{k,0:k-1}) \propto \bar{\pi}_k(s_k).$$

Pomocnou funkci  $p(s_{k-1}, s_k)$  potřebujeme umět vyčíslit s.v.v  $q(ds_{k-1}, ds_k)$ , tedy pro  $s_{k,0:k-1} = s_{k-1,0:k-1}$ , což vede na váhovou funkci s.v.v  $q(ds_{k-1}, ds_k)$ :

$$w(s_{k-1}, s_k) = \frac{\gamma_k(s_{k,k}|s_{k,0:k-1})}{q_k(s_{k,k}|s_{k,0:k-1})} \frac{\bar{\gamma}_{k-1}(s_{k-1,0:k-1})}{q_{k-1}(s_{k-1,0:k-1})} = \alpha_k(s_{k-1}, s_k) w_{k-1}(s_{k-1}).$$

Inkrementální váha je v tomto případě  $\alpha_k(s_{k-1}, s_k) = \frac{\gamma_k(s_{k,k}|s_{k,0:k-1})}{q_k(s_{k,k}|s_{k,0:k-1})}$  (speciálně pro HMM

$$\alpha_k(x_{k-1}, x_k) = \frac{p(y_k|x_k)p(x_k|x_{k-1})}{q_k(x_k|x_{0:k-1})}).$$

Po marginalizaci získáme opět vzorky  $X_k^i \sim q_k(s_k) = (q_{k-1}Q_k)(s_k)$  a váhy  $W_k^i \propto \alpha_k(S_{k-1}^i, S_k^i)W_{k-1}^i$ . Vážené vzorky  $\{W_k^i, X_k^i\}$  budou díky konstrukci řádně vážené podle  $\pi_k(x_k)$  (a lze je využít v následujícím kroku metody).

Opět jsme tedy získali algoritmus PF, tentokrát interpretovaný pomocí protahování trajektorií. Zároveň jsme opět ukázali, na jaké obecnější modely můžeme takový algoritmus využít - modely se známým podmíněním historií (známá  $\pi(x_k|x_{0:k})$ ).

#### 4.2.5 Zpětná jádra

V případě, že nemáme z modelu k dispozici přechodové jádro jako u PF, ale umíme bodově vyčíslit každou  $\pi_k(s_k)$ , musíme pomocné hustoty  $p(s_{k-1}, s_k)$  konstruovat uměle. Stále požadujeme, aby  $\pi_k(s_k)$  byla jejich magrinálou, což vede k následující intuitivní konstrukci.

Zvolíme libovolné zpětné jádro  $L_{k-1}(s_k, s_{k-1})$  (zpozorňujeme, že podle def. 4.4 je  $L_{k-1}$  rozdělením pravděpodobnosti v proměnné  $s_{k-1}$ ) a pomocné sdružené rozdělení vyjádříme ve tvaru:

$$p(s_{k-1}, s_k) = \pi_k(s_k) L_{k-1}(s_k, s_{k-1}).$$

Díky tomu, že  $L_{k-1}$  je zpětné jádro (podle def. 4.4 je rozdělením pravděpodobnosti v proměnné  $s_{k-1}$ ), platí požadovaný vztah pro aplikaci lemma 4.5 :  $\int p(s_{k-1}, s_k) ds_{k-1} = \pi_k(s_k)$ .

Konstrukce vede na vzorec pro výpočet váhové funkce:

$$w_k(s_{k-1}, s_k) = \frac{\pi_k(s_k) L_{k-1}(s_k, s_{k-1})}{q_{k-1}(s_{k-1}) Q_k(s_{k-1}, s_k)} = \frac{1}{Z_k} \frac{\gamma_k(s_k) L_{k-1}(s_k, s_{k-1})}{\gamma_{k-1}(s_{k-1}) Q_k(s_{k-1}, s_k)} \frac{\gamma_{k-1}(s_{k-1})}{q_{k-1}(s_{k-1})}$$

a inkrementální váhy

$$\alpha_k(s_{k-1}, s_k) = \frac{\gamma_k(s_k) L_{k-1}(s_k, s_{k-1})}{\gamma_{k-1}(s_{k-1}) Q_k(s_{k-1}, s_k)}. \quad (14)$$

Optimální volba zpětného jádra [5] (ve smyslu minimalizace rozptylu vah)

$$L_{k-1}^{opt}(s_k, s_{k-1}) := \frac{q_{k-1}(s_{k-1}) Q_k(s_{k-1}, s_k)}{q_k(s_k)} \quad (15)$$

vede na váhovou funkci

$$w_k^{opt}(s_{k-1}, s_k) := \frac{\gamma_k(s_k)}{q_k(s_k)}.$$

Optimální volbou by tedy bylo využít pro výpočet vah návrhovou hustotu (12). Tu ovšem krom možného provedení odhadu (13) neumíme vyčíslit.

Další suboptimální jádra můžeme dostat např. záměnou  $q_{k-1}(s_{k-1}) \leftarrow \pi_{k-1}(s_{k-1})$ , která tvar optimálního zpětného jádra (15) a příslušné inkrementální váhy upraví na:

$$\begin{aligned} L_{k-1}^s(s_k, s_{k-1}) &:= \frac{\pi_{k-1}(s_{k-1})Q_k(s_{k-1}, s_k)}{(\pi_{k-1}Q_k)(s_k)}, \\ \alpha_k^s(s_{k-1}, s_k) &:= \frac{\gamma_k(s_k)}{\int \gamma_{k-1}(s_{k-1})Q_k(s_{k-1}, s_k)ds_{k-1}} \end{aligned} \quad (16)$$

Zde můžeme využít i přesnější metody pro výpočet návrhové hustoty, jelikož  $\gamma_{k-1}(s_{k-1})$  umíme bodově vyčíslit.

V případě, že n.v.  $S_k$  sdílí obor hodnot ( $R(S_k) = R(S_{k-1}); \forall k \in \mathbb{N}$ ), můžeme volit přechodové jádro  $Q_k(s_{k-1}, s_k)$  se stabilním rozdělením  $\pi_k(s_k)$ . Metodami konstrukce takových jader se zabývá teorie MCMC (my jsme uvedli např. Metropolisovo-Hastingsovo nebo Gibbsovo jádro v sekci 2.3). Můžeme-li se domnívat, že si budou po sobě jdoucí cílová rozdělení podobná ( $\pi_{k-1}(s_{k-1}) \approx \pi_k(s_k)$ ). Podobnost přeneseme pomocí vztahu

$$(\pi_{k-1}Q_k)(s_k) \approx (\pi_kQ_k)(s_k) = \pi_k(s_k)$$

na suboptimální zpětné jádro (16):

$$L_{k-1}^m(s_k, s_{k-1}) := \frac{\pi_{k-1}(s_{k-1})Q_k(s_{k-1}, s_k)}{\pi_k(s_k)} \quad (17)$$

a inkrementální váhu

$$\alpha_k^m(s_{k-1}, s_k) := \frac{\gamma_k(s_k)}{\gamma_{k-1}(s_{k-1})}. \quad (18)$$

Tento postup nám umožní inkrementální váhu bodově vyčíslit.

V [5] je zdůrazňována potřeba volit zpětné jádro co nejpodobnější  $L_{k-1}^{opt}$ , jinak se algoritmy stanou neefektivními. Ve zmíněné práci jsou také uvedeny konvergenční výsledky pro algoritmy využívající zpětná jádra.

#### 4.2.6 Obecný sekvenční Monte Carlo algoritmus

Obecný SMC algoritmus je velmi podobný algoritmu 5. Liší se jen ve značení n.v. (za kterou zde považujeme obecnější objekt), což se také promítne do tvaru návrhové přechodové hustoty a inkrementální váhy.

Mimo hledanou aproximaci získáme v kroku  $k$  také aproximaci  $\hat{Z}_k \approx \int \gamma_k(s_k)ds_k$  pomocí sekvenčního napočítávání poměrů

$$\frac{Z_k}{Z_{k-1}} = \frac{\int \gamma_k(s_k)ds_k}{\int \gamma_{k-1}(s_{k-1})s_k} \approx \sum W_{k-1}^i \alpha_k(S_{k-1}^i, S_k^i).$$

**Algoritmus 6** Sekvenční Monte Carlo s adaptivním převzorkováním II**Pro**  $k = 1$ **Vygeneruj** vzorky  $S_0^i \sim q_0(s_0)$  a spočti váhy  $W_0^i \propto w(S_0^i)$ **Pro**  $k > 1$ ;**Je-li**  $ESS < \frac{N}{2}$ ; Převzorkuj  $S_{k-1}^{1:N}$  na soubor vzorků i.i.d. podle  $\pi_{k-1}(s_{k-1})$  $\forall i = 1 : N$ **Vytvoř** vzorek  $S_k^i \sim q_k(s_k | S_{k-1}^i)$ **Vypočti** váhy  $W_k^i \propto \alpha_k(S_{k-1}^i, S_k^i) W_{k-1}^i$ **Výstupem** kroku  $k$  je aproximace  $\hat{\pi}^N(ds_k)$  (soubor vážených vzorků).**4.2.7 Optimální vyhlazování**

K řešení úlohy 3.4 využijeme techniku složených filtrů. Obecně lze řešení rozložit do dvou kroků. Získání aproximace  $p(x_k | y_{1:k}); \forall k \in [0, T]$  pomocí částicových filtrů (řešení je popsáno v sekci 4.1). V druhém kroku sestavíme úlohu filtrace pro tzv. zpětný filtr. To provedeme takovým způsobem, abychom pro řešení zpětné filtrace mohli opět použít částicové filtry. Tímto postupem získáme efektivní algoritmus (v porovnání s např. MCMC) pro řešení úlohy optimálního vyhlazování.

Konstrukci zpětného filtru můžeme provést různě. Uvedeme příklad, který vychází z následující faktorizace sdruženého rozdělení HMM [8]

$$\begin{aligned}
 p(x_{0:T} | y_{1:T}) &= p(x_T | y_{1:T}) \prod_{k=0}^{T-1} p(x_k | x_{k+1}, y_{1:T}) \\
 &= p(x_T | y_{1:T}) \prod_{k=0}^{T-1} p(x_k | x_{k+1}, y_{1:n})
 \end{aligned}$$

kde

$$p(x_k | x_{k+1}, y_{1:k}) = \frac{p(x_{k+1} | x_k) p(x_k | y_{1:k})}{p(x_{k+1} | y_{1:k})} \quad (19)$$

s úmluvou, že  $p(x_0 | y_{1:0})$  chápeme jako apriorní rozdělení skrytého procesu.

Pro získání vzorků z hledaného rozdělení využijeme zpětný filtr (začneme se vzorky z rozdělení  $p(x_T | y_{1:T})$ , budeme vzorkováním s přechodovou hustotou  $T(X_{k+1}^i, x_k) = p(x_k | X_{k+1}^i, y_{1:k})$ .

Přímým dosazením do formule (19) získáme předpis pro aproximaci hledaného rozdělení (a metodu ‘Forward-filtering backward sampling’):



$$\begin{aligned}\hat{p}^N(dx_k|X_{k+1}, y_{1:T}) &= \frac{p(X_{k+1}|dx_k)\hat{p}^N(dx_k|y_{1:k})}{\int p_{X_{k+1}|X_k}(X_{k+1}|\xi)\hat{p}_{X_k|Y_{1:k}}^N(\xi|y_{1:k})d\xi} \\ &= \sum_{i=1}^N \frac{W_k^i p(X_{k+1}|X_k^i)}{\sum_{j=1}^N W_k^j p(X_{k+1}|X_k^j)} \delta_{X_k^i}(dx_k).\end{aligned}$$

Všimněme si, že zpětný filtr můžeme formálně popsat také výběrem návrhového přechodového rozdělení  $q(dx_{k-1}|x_k) = \hat{p}^N(dx_k|y_{1:k})$  a váhovou funkcí pro pomocný sdružený vzorek:

$$w(x_k, x_{k-1}) = \frac{W_k^i p(x_{k+1}|X_k^i)}{\sum_{j=1}^N W_k^j p(x_{k+1}|X_k^j)}.$$

Další metody řešení úlohy 3.4 jsou uvedeny v [8]. Princip je ale u všech obdobný - řešení úlohy pomocí dvou následných úloh filtrace, kde výsledek první využijeme pro vyjádření parametrů (návrhového rozdělení a váhové funkce) druhé- zpětné filtrace.

### 4.3 Úpravy algoritmů

V této podkapitole uvedeme několik úprav SMC algoritmů. Společným jmenovatelem je snaha snížit rozptyl vah tvorbou vzorků blíže optimálnímu návrhovému rozdělení.

#### 4.3.1 Pomocné filtry

Jako jediná z úprav je vázána na úlohu optimální filtrace pro HMM. Samotná idea pomocných filtrů [8] je ale obecnější (PF jsme odvozovali pomocí pomocných sdružených rozdělení). Pomocné filtry ['auxiliary particle filter'] byly původně popsány pomocí pomocných proměnných a odtud získali svůj název.

Podstatou metody je zefektivnění kroku převzorkování. Jistě bychom v kroku  $k$  neradi zahazovali vzorky, které budou v oblastech s vysokou pravděpodobností v kroku  $k+1$ . Pomocné filtry se snaží aproximovat hustotu  $p(y_{k+1}|x_k)$  a vzít tuto informaci v úvahu při převzorkování.

Jednoduše je lze popsat jako particle filtr s cílovým rozdělením

$$\tilde{\pi}(x_{0:k}) \propto p(x_{0:k}|y_{1:k})\tilde{p}(y_{k+1}|x_k),$$

kde  $\tilde{p}(y_{k+1}|x_k)$  volíme jako aproximaci hustoty  $p(y_{k+1}|x_k)$ , pokud ji neznáme analyticky.

Inkrementální váha pro takový filtr má tvar

$$\alpha_k(x_{0:k}) = \frac{\tilde{\gamma}_k(x_{0:k})}{\tilde{\gamma}_{k-1}(x_{0:k-1})q_k(x_k|x_{0:k-1})} = \frac{p(y_k|x_k)p(x_k|x_{k-1})\tilde{p}(y_{k+1}|x_k)}{q_k(x_k|x_{0:k-1})\tilde{p}(y_k|x_{k-1})}.$$

V případě, že chceme s těmito výsledky aproximovat funkcionál (1), využijeme získanou aproximaci  $\hat{\hat{\pi}}$ , kde

$$\tilde{\pi}_{k-1}(x_{0:k-1})q(x_k|x_{0:k-1}) = \tilde{p}(x_{1:k-1}|y_{1:k})q(x_k|x_{0:k-1}),$$

jako návrhové rozdělení pro IS odhad s váhami

$$\tilde{w}(x_{0:k}) = \frac{p(x_{0:k}|y_{1:k})}{\tilde{\gamma}_{k-1}(x_{0:k-1})q(x_k|x_{0:k-1})} = \frac{p(y_k|x_k)p(x_k|x_{k-1})}{\tilde{p}(y_k|x_{k-1})q(x_k|x_{0:k-1})}.$$

V případě volby optimálních

$$\begin{cases} q(x_k|x_{0:k-1}) = & p(x_k|y_k, x_{k-1}) \\ \tilde{p}(y_{k+1}|x_k) = & p(y_{k+1}|x_k) \end{cases}$$

získáme tzv. perfektní adaptaci, kdy

$$\alpha_k(x_{0:k}) = p(y_k|x_{k-1})$$

a  $\tilde{w}(x_{0:k}) = 1$ .

### 4.3.2 Resample-move

Převzorkování způsobí zahození několika částic a multiplikaci ostatních - obecně degeneraci souboru vzorků. Myšlenkou v pozadí resample-move [8] je přinést do souboru vzorků určité oživení (duplikované vzorky mají negativní vliv na rozptyl odhadu funkcionálu (1)). Idea této metody je posunout v kroce  $k$  celý soubor podle přechodového jádra  $K(s_k, s_k^*)$  se stabilním rozdělením  $\pi_k(s_k^*)$ .

Po vytvoření nových vzorků a výpočtu vah, provedeme převzorkování. Získáme soubor vzorků  $S_k^{1:N}$  i.i.d. podle  $\pi_k(s_k)$ . Po posunutí podle takto zvoleného jádra budou výsledné vzorky opět i.i.d. podle  $\pi_k(s_k)$ .

Několik metod konstrukce takovýchto požadovaných jader jsme uvedli v kapitole 2 (Metropolisova-Hastingsova jádra, Gibbsova jádra).

### 4.3.3 Move-reweight

Tato úprava vznikla rozpracováním úpravy MCMC posunutí [15]. Její smysl není v oživení degenerovaného souboru, ale oddálení kroku převzorkování. Metoda resample-move vyžaduje vzorky i.i.d. z požadovaného rozdělení. V případě, že máme soubor vzorků  $S^i \sim q(s)$  a propagujeme jej podle jádra  $K(s, s^*)$  se stabilním rozdělením  $\pi(s^*)$ , budou výsledné vzorky ‘blíže’ požadovanému rozdělení (to je základem algoritmů MCMC).

Mějme vážené vzorky  $\{W^i, S^i\} \sim \pi(s); S^i \sim q(s)$  a vytvořme nový soubor  $S^{*i} \sim (qK)(s^*)$ . Pokud na nové vzorky nahlížíme jako na vzorky z návrhového rozdělení  $(qK)$ , musíme pro získání řádně váženého souboru ještě napočítat váhy  $W^{*i}$ . K tomu můžeme využít obdobného postupu jako v sekci 4.2.3 a 4.2.5. Zvolíme zpětné jádro  $L(s^*, s)$  a vytvoříme pomocné rozdělení  $p(s, s^*) = \pi(s^*)L(s^*, s)$ .

Abychom získali řádně vážené vzorky, napočítáme nové váhy ze vztahu:

$$w(s, s^*) = \frac{p(s, s^*)}{q(s)K(s, s^*)} = \frac{\pi(s^*)L(s^*, s)}{q(s)K(s, s^*)}.$$

Optimální volba zpětného jádra zůstává stejná jako v sekci 4.2.5, tedy (15).

V publikaci [15] jsou dále zmíněny další volby:

$$L_1(s, s^*) := \frac{\pi(s)K(s, s^*)}{\pi(s^*)}$$

$$L_2(s, s^*) := K(s^*, s)$$

$$L_3(s, s^*) := q(s)$$

které postupně vedou na váhové funkce:

$$w_1(s, s^*) = \frac{\pi(s)}{q(s)} = w(s)$$

$$w_2(s, s^*) = \frac{\pi(s^*)K(s^*, s)}{q(s)K(s, s^*)}$$

$$w_3(s, s^*) = \frac{\pi(s^*)}{K(s, s^*)}.$$

Publikace [15] dále popisuje kvalitativní aspekty těchto voleb.

Výhodou tohoto postupu je možnost aplikovat jej dokonce i v případě, kdy  $K$  není jádrem s vhodným stabilním rozdělením, ale jen jeho aproximací. Pak se stále můžeme domnívat, že posunuté vzorky budou v daném smyslu blíže cílového rozdělení (pokud tedy máme dobrou aproximaci).

## 5 Aplikace

V této kapitole uvedeme několik příkladů použití sekvenčních algoritmů.

### 5.1 Aproximace aposteriorního rozdělení statického parametru

Budeme se zabývat řešením úlohy 3.1.

Předpokládejme, že máme k dispozici sérii  $m$  měření skrytého parametru. Označme tento soubor  $Y_{1:m}$ . Známe-li model pozorování

$$Y_k \sim p(y_k|\theta) = p(y_k|\theta, y_{1:k-1}), \quad (20)$$

můžeme zvolit vhodné apriorní rozdělení  $p(\theta)$  a formulovat úlohu aproximace série aposteriorních rozdělení

$$\pi_k(\theta) = p(\theta|y_{1:k}). \quad (21)$$

Úlohu budeme řešit MC algoritmy, které vyžadují možnost vyjádřit cílové hustoty ve tvaru (2). Pro každé  $k \in \mathbb{N}$  můžeme cílovou hustotu rozložit do tvaru:

$$\pi_k(\theta) = \frac{p(y_{1:k}|\theta)p(\theta)}{p(y_{1:k})} = \frac{\gamma_k(\theta)}{Z_k}. \quad (22)$$

Pro řešení úlohy pomocí sekvenčních algoritmů ukážeme, že lze problém přeformulovat na problém aproximace skrytého procesu.

**Úloha 5.1.** (aposteriorní rozdělení skrytého parametru)

Hledáme aproximaci aposteriorních rozdělení (21). Pro model:

- Data  $Y_{1:m}$  jsou i.i.d.  $p(y_k|\theta) = \mathcal{N}(y_k; \theta, \sigma^2)$ , kde  $\mathcal{N}$  je normální rozdělení a  $\theta = 3$  a  $\sigma^2 = 1$  jsou parametry modelu pozorování.
- Zvolená apriorní hustota  $p_a(\theta) = \mathcal{N}(\theta; 0, 100)$ .

Všechny testované algoritmy jsou aplikovány na společná data  $Y_{1:m}$ ;  $m = 25$  a k aproximaci využívají stejný počet vzorků  $N = 250$ . Takto můžeme porovnat výsledky zpracování stejného souboru dat různými metodami.

#### 5.1.1 Degenerovaný proces

Jednou možností je úlohu formálně převést na úlohu optimální filtrace a řešit pomocí efektivních PF algoritmů. Na statický parametr můžeme nahlížet jako na stacionární proces na  $\theta_k|_{k \in \mathbb{N}_0}$ , s rozdělením přechodu, které označíme:

$$T(\theta_{k-1}, d\theta_k) := p(d\theta_k|\theta_{k-1}) = \delta_{\theta_{k-1}}(d\theta_k). \quad (23)$$

Protože nyní je skrytý proces jednoznačně určen počátečním stavem, řekneme o něm, že je degenerovaný.

Náhodné procesy  $(X_k)_{k \in \mathbb{N}_0}, (Y_k)_{k \in \mathbb{N}}$  tak budou společně tvořit HMM popsany rozděleními

$$\begin{cases} p_a(\theta) & ; \text{ apriorní rozdělení } \theta \\ p(y_k|\theta_k) & ; \text{ model pozorování(20) } \\ p(d\theta_k|\theta_{k-1}) & ; \text{ model přechodu } \end{cases}$$

Pro aplikaci algoritmů PF potřebujeme bodově vyjádřit nenormalizovanou váhovou funkci. V algoritmu 4 proto budeme volit návrhové přechodové jádro

$$Q_k(\theta_{k-1}, d\theta_k) = \delta_{\theta_{k-1}}(d\theta_k).$$

Jelikož  $T_k \ll Q_k$  stačí nám pro obecné  $k$  umět vyčíslit váhovou funkci sdruženého vektoru n.v. s.v.v  $q_k(\theta_{0:k})$  (tj.  $\forall i, j \leq k : \theta_i = \theta_j$ ) ve tvaru

$$w(\theta_{0:k}) \propto \frac{p(\theta_0)}{q_0(\theta_0)} \prod_{n=1}^k p(y_n|\theta_n) = \frac{p(\theta_0)}{q_0(\theta_0)} \prod_{n=1}^k p(y_n|\theta_0), \quad (24)$$

Díky tvaru výrazu napravo ve vztahu (24) si můžeme všimnout, že touto formulací fakticky dostáváme úlohu, kterou můžeme řešit algoritmem IS. K tomu jsme mohli dojít i přímou aplikací IS na rozdělení (21). Takhle ale můžeme využít sekvenční formulace úlohy.

Problém budeme řešit algoritmem 4 (SIS). K tomu potřebujeme ještě znát návrhovou hustotu  $q_0(\theta_0)$  pro vytvoření vážených vzorků z  $p(\theta_0) = p_a(\theta_0)$ , kterou ovšem už můžeme volit téměř libovolně v souladu s IS metodami (požadujeme  $p(d\theta_0) \ll q_0(d\theta_0)$ ).

Inkrementální váhu dostaneme přímo z tvaru váhy sdruženého vzorku (24) :

$$\alpha_k(\theta_{0:k}) = p(y_k|\theta_k) = \mathcal{N}(y_k; \theta_k, s_{2y}).$$

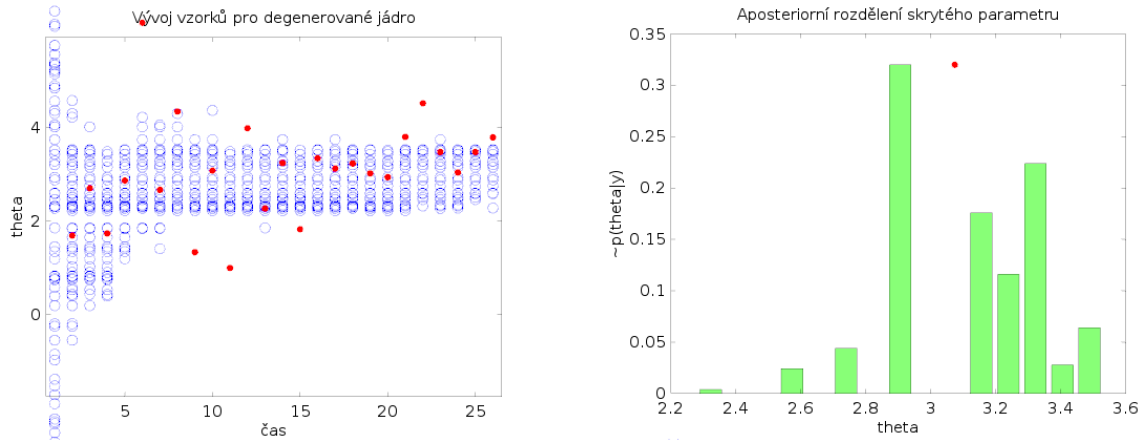
Tento algoritmus bude trpět stejným problémem jako IS a to složitostí volby návrhového rozdělení (zde  $q_0(\theta_0)$ ), které by minimalizovalo rozptyl vah. Při numerickém řešení budeme volit

$$q_0(\theta_0) = p_a(\theta_0) = \mathcal{N}(\theta_0; 0, 100),$$

což nám také usnadní implementaci, jelikož z  $p_a(\theta)$  voleného v tého úloze umíme snadno vzorkovat. Výsledky simulace jsou zobrazeny na obrázcích 1(a,b).

Záměrně jsme také zmínili řešení pomocí algoritmu SIS, který neobsahuje krok převzorkování. V uvedeném případě by duplikace vzorků pouze navýšila výpočetní náročnost. Jistou formu převzorkování ovšem zahrnout můžeme, pokud budeme v průběhu běhu algoritmu průběžně zahazovat vzorky s nízkými vahami. Tento postup bude mít pozitivní vliv na výpočetní složitost. V jeho průběhu ale dojde k degeneraci sady vzorků, což se negativně projeví na konvergenci vznikajících aproximací. Velkou výhodou postupu ovšem zůstává fakt, že pro výpočet inkrementální váhy potřebujeme mít k dispozici pouze poslední pozorování a ne všechna, jak tomu bude u dále uvedených postupů.

Pokud by se povedlo odstranit problém s degenerací vzorků (např. kombinací postupů), je SIS kandidátem na rychlý algoritmus pro nasazení na odhady v real-time provozu.



(a) Vývoj navzorkovaných částic v různých časech při aplikaci postupu v sekci 5.1.1 pro řešení úlohy 5.1. Modré kružnice představují vzorky vniklé převzorkováním váženého souboru  $\{W_k^{1:N}, \theta_k^{1:N}\} \sim p(\theta|y_{1:k})$  a plně červené kruhy představují pozorované hodnoty v příslušném čase.

(b) Histogram vzorků vniklých převzorkováním váženého souboru  $\{W_k^{1:N}, \theta_k^{1:N}\} \sim p(\theta_k|y_{1:k})$  při aplikaci postupu v sekci 5.1.1 pro řešení úlohy 5.1. Červená tečka reprezentuje na vodorovné ose polohu výběrového průměru datového souboru.

Obrázek 1: Sekvenční aproximace posteriorního rozdělení, degenerovaný proces.

### 5.1.2 Nezávislá návrhová rozdělení

Další možností, jak sekvenčně získávat aproximace cílových rozdělení v úloze 5.1, je sekvenční aplikace algoritmu IS. Pokud bychom chtěli využít metodu propagace částic (viz. sekce 4.2.2) ( $\theta_k^i \sim q_k(\theta_k|\theta_{k-1}^i)$ ), musíme umět nějakým způsobem napočítat váhy (12):

$$W_k^i \propto \frac{p(\theta_k^i|y_{1:k})}{(q_{k-1}Q_k)(\theta_k^i)}.$$

V sekci 4.2.2 jsme zmínili, že tuto váhu obecně neumíme exaktně vyčíslit - pouze odhadnout pomocí vztahu (13). Tento problém ale můžeme obejít speciální volbou návrhového jádra  $Q_k(\theta_{k-1}, \theta_k)$  ve formě tzv. nezávislého návrhu

$$Q_k(\theta_{k-1}, \theta_k) = q_k(\theta_k).$$

Od nezávislého návrhového rozdělení požadujeme možnost jej bodově vyčíslit. Ve smyslu minimalizace rozptylu odhadu výsledného funkcionálu jej můžeme volit jako aproximaci  $\pi_k(\theta_k)$  a jeho případné parametry (např. střední hodnotu a rozptyl) odhadovat z aproximace  $\pi_{k-1}^N$  z předchozího kroku. Váhy v každém kroku  $k$  vyjádříme pomocí vztahu:

$$W_k^i \propto \frac{p(\theta_k^i|y_{1:k})}{q_k(\theta_k^i)} = \frac{p_a(\theta_k^i)}{q_k(\theta_k^i)} \prod_{n=1}^k p(y_{1:k}|\theta_k^i).$$

Nezávislá návrhová hustota je při numerickém experimentu volena ve formě normálního rozdělení  $\mathcal{N}(\theta_k; \mu_k, \sigma_k^2)$  se střední hodnotou  $\mu_k = \sum_{i=1}^N W_{k-1}^i \theta_{k-1}^i$  (tj. střední hodnoty souboru v předcházejícím

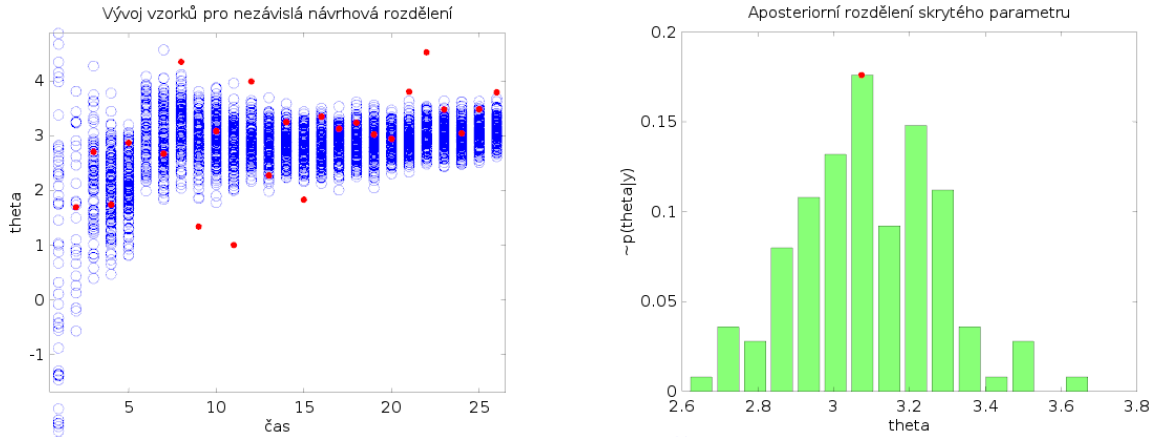
kroku) a rozptylem  $\sigma_k^2 = 4 * \sum_{i=1}^N W_{k-1}^i (\theta_{k-1}^i - \mu_k)^2$  (tj. 4-násobek váženého momentového odhadu rozptylu z minulého kroku).

Úlohu řešíme algoritmem 6 pro parametry

$$\begin{aligned} q_0(\theta_0) &= p_a(\theta_0) = \mathcal{N}(\theta_0; 0, 100) \\ w(\theta_0) &= \frac{p_a(\theta_0)}{q_0(\theta_0)} = 1 \\ q_k(\theta_k | \theta_{k-1}) &= q_k(\theta_k) = \mathcal{N}(\theta_k; \mu_k, \sigma_k^2) \\ \alpha_k(\theta_{k-1}^i, \theta_k) W_{k-1}^i &\propto w_k(\theta_k) = \frac{\gamma_k(\theta_k)}{q_k(\theta_k)} \end{aligned}$$

Poznámka : Obešli jsme vztah pro výpočet inkrementální váhy. Ta by totiž měla ve jmenovateli výraz  $\gamma_{k-1}(\theta_{k-1}^i)$ , což z povahy numerického řešení může nabývat i nulových hodnot. Jedná se ovšem o odstranitelnou singularitu.

Výsledky simulace jsou zobrazeny na obrázcích 2(a,b).



(a) Vývoj navzorkovaných částic v různých časech při aplikaci postupu v sekci 5.1.2 pro řešení úlohy 5.1. Modré kružnice představují vzorky vniklé převzorkováním váženého souboru  $\{W_k^{1:N}, \theta_k^{1:N}\} \sim p(\theta | y_{1:k})$  a plné červené kruhy představují pozorované hodnoty v příslušném čase.

(b) Histogram vzorků vniklých převzorkováním váženého souboru  $\{W_k^{1:N}, \theta_k^{1:N}\} \sim p(\theta_k | y_{1:k})$  při aplikaci postupu v sekci 5.1.2 pro řešení úlohy 5.1. Červená tečka reprezentuje na vodorovné ose polohu výběrového průměru datového souboru.

Obrázek 2: Sekvenční aproximace posteriorního rozdělení; nezávislé návrhové hustoty.

### 5.1.3 Zpětná jádra

V posledním prezentovaném případě využijeme zpětná jádra uvedená v sekci 4.2.5. Nové vzorky opět budeme tvořit metodou propagace s využitím návrhového jádra  $Q_k(\theta_{k-1}, \theta_k)$ , které bude v

tomto případě záviset na předchozím vzorku a nebude degenerované. Pro vyčíslení inkrementální váhy ze vztahu (14) potřebujeme zvolit vhodné zpětné jádro  $L_{k-1}(\theta_k, \theta_{k-1})$ . Využijeme možnosti bodového vyčíslení  $\gamma_k(\theta_k)$  v (22) a za  $Q_k(\theta_{k-1}, \theta_k)$  budeme volit Metropolisovo-Hastinsovo jádro (4). Pro takto vnořený MCMC algoritmus musíme ještě zvolit návrhové rozdělení, které označíme  $q_k^{mcmc}(\theta^*|\theta)$ .

Volíme  $q_k^{mcmc}(\theta^*|\theta) = \mathcal{N}(\theta^*; \theta, 0.01)$ , což díky symetrii vede na pravděpodobnost přijetí vzorku  $A(\theta^*|\theta) = \min \left\{ 1, \frac{\gamma_k(\theta^*)}{\gamma_k(\theta)} \right\}$  (Metropolisův algoritmus).

Tato konstrukce zpětného jádra nám umožní vyčíslit inkrementální váhu ze vztahu (18)

$$\alpha_k(\theta_{k-1}, \theta_k) = \frac{\gamma_k(\theta_{k-1})}{\gamma_{k-1}(\theta_{k-1})}.$$

Úlohu řešíme algoritmem 6 pro parametry

$$\begin{aligned} q_0(\theta_0) &= p_a(\theta_0) = \mathcal{N}(\theta_0; 0, 100) \\ w(\theta_0) &= \frac{p_a(\theta_0)}{q_0(\theta_0)} = 1 \\ q_k(\theta_k|\theta_{k-1}) &= Q_k(\theta_{k-1}, \theta_k) \text{ (vnořený MCMC)} \\ q_k^{mcmc}(\theta^*|\theta) &= \mathcal{N}(\theta^*; \theta, 0.01) \\ \alpha_k(\theta_{k-1}, \theta_k) &= \frac{\gamma_k(\theta_{k-1})}{\gamma_{k-1}(\theta_{k-1})} \end{aligned}$$

Výsledky simulace jsou zobrazeny na obrázcích 3(a,b).

Použité algoritmy můžeme při řešení obecného problému kombinovat. Výhodné se jeví např. využít metodu s degenerovanými jádry pro rychlé zpracování dat, kterou bychom v případě poklesu kvality vzorků mohli prostřídat pár kroky sofistikovanějších metod pro oživení souboru vzorků (např. zmíněné nezávislé návrhové hustoty nebo některé metody ze sekce 4.3).

## 5.2 Dynamické deterministické systémy

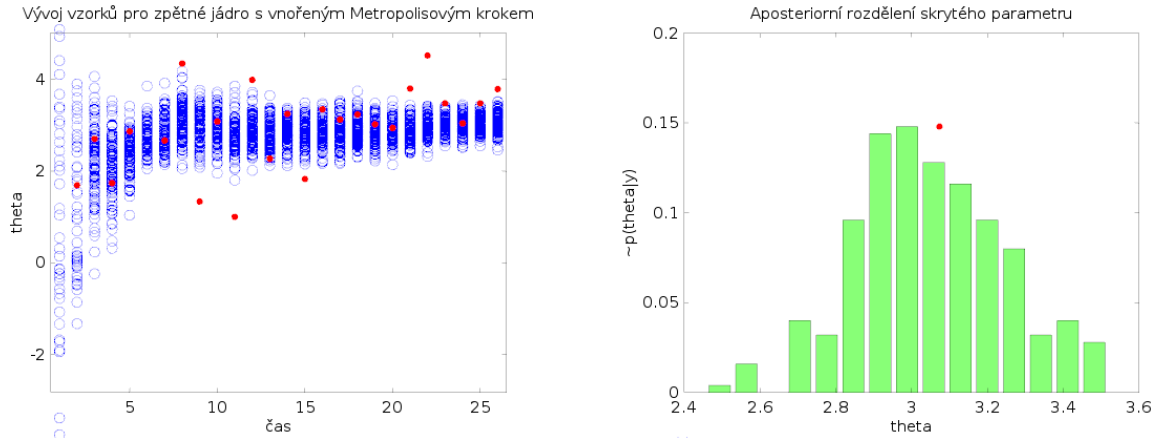
Sekvenční Monte Carlo algoritmy se přirozeně aplikují na stochastické procesy. V této kapitole se budeme zabývat deterministickými systémy. Pomocí algoritmu pro optimální filtraci se nejprve pokusíme odhadnout stav nepřímo pozorovaného systému.

V poslední části sekce se budeme zabývat bayesovským odhadem parametru pohybu deterministického systému pozorovaného skrze zašuměná data.

### 5.2.1 Matematické kyvadlo

Časový vývoj odchylky  $u$  matematického bez tlumení můžeme popsat počáteční úlohou tvaru





(a) Vývoj navzorkovaných částic v různých časech při aplikaci postupu v sekci 5.1.3 pro řešení úlohy 5.1. Modré kružnice představují vzorky vniklé převzorkováním váženého souboru  $\{W_k^{1:N}, \theta_k^{1:N}\} \sim p(\theta|y_{1:k})$  a plně červené kruhy představují pozorované hodnoty v příslušném čase.

(b) Histogram vzorků vniklých převzorkováním váženého souboru  $\{W_k^{1:N}, \theta_k^{1:N}\} \sim p(\theta_k|y_{1:k})$  při aplikaci postupu v sekci 5.1.3 pro řešení úlohy 5.1. Červená tečka reprezentuje na vodorovné ose polohu výběrového průměru datového souboru.

Obrázek 3: Sekvenční aproximace aposteriorního rozdělení; zpětná jádra.

$$\begin{cases} \frac{\partial^2}{\partial t^2} u(t) + \frac{g}{l} \sin(u(t)) = 0 & ; t \in \mathbb{R}^+ \\ u(0) = u_0 \\ u'(0) = v_0 \end{cases}, \quad (25)$$

kde  $g$  a  $l$  reprezentují parametry úlohy (tíhové zrychlení a délku závěsu resp.).

Pro malé hodnoty výchylky  $u(t)$  lze využít aproximace  $\sin(u(t)) \approx u(t)$ . Vložením této aproximace do rovnice (25) a přeznačením  $\omega^2 = \frac{g}{l}$  získáme jednodušší obyčejnou lineární diferenciální rovnici

$$\begin{cases} \frac{\partial^2}{\partial t^2} u(t) + \omega^2 u(t) = 0 & ; t \in \mathbb{R}^+ \\ u(0) = u_0 \\ u'(0) = v_0 \end{cases}, \quad (26)$$

jejíž řešení umíme explicitně vyjádřit ve tvaru

$$\begin{cases} u(t) = \frac{v_0}{\omega} \sin(\omega t) + u_0 \cos(\omega t) & ; t \in \mathbb{R}^+ \\ u'(t) = v_0 \cos(\omega t) - \omega u_0 \sin(\omega t) & ; t \in \mathbb{R}^+ \end{cases}.$$

### 5.2.2 Sledování kyvadla

V této sekci budeme řešit úlohu sledování pro simulovaný experiment, ve kterém data pochází ze zašuměného měření stavu deterministického evolučního procesu. Vývoj skrytého procesu  $u(t)$

popisuje aproximační rovnice matematického kyvadla (26) s parametry

$$\begin{cases} \omega^2 = 9.81 \\ u_0 = 0.1 \\ v_0 = 0 \end{cases}.$$

Simulovaná data pocházejí z rozdělení  $\mathcal{N}(y_k; u(t_k), 0.0036)$ , kde  $t_k$  označíme čas provedení měření  $y_k$  a  $u(t_k)$  je výchylka simulovaného vnitřního procesu ve stejném čase.

Pro formulaci úlohy jako problému 3.3 (optimální filtrace) budeme skrytý proces modelovat přibližným stochastickým modelem. Uvažujeme lineární gaussovský model skrytého procesu  $U_k|_{k \in \mathbb{N}_0}$ :

$$\begin{cases} U_{k+1} = U_k + V_k; V_k \sim \mathcal{N}(0, \sigma_{V_k}^2) & ; k \in \mathbb{N}_0 \\ U_0 \sim p_a(u_0) = \mathcal{N}(u_0; 0, 0.1) \end{cases}$$

Tento model je markovský s přechodovou hustotou  $p(u_{k+1}|u_k) = \mathcal{N}(u_{k+1}; U_k, \sigma_{V_k}^2)$ .  $\sigma_{V_k}$  zvolíme závislé na časových rozestupech měření ve tvaru  $\sigma_{V_k} = 0.5\Delta t = \frac{5}{2m}$ .

Mějme soubor dat  $Y_{1:m}$ . V tomto experimentu předpokládáme, že model pozorování známe a nabývá tvaru:

$$p(y_k|u_{0:k}, y_{1:k-1}) = p(y_k|u_k) = \mathcal{N}(y_k; u_k, 0.0036).$$

Procesy  $U_k|_{k=0:m}, Y_k|_{k \in 1:m}$  tvoří konečný HMM se sdruženým rozdělením (7):

$$p(u_{0:m}, y_{1:m}) = p(u_0) \prod_{k=1}^m p(y_k|u_k)p(u_k|u_{k-1}).$$

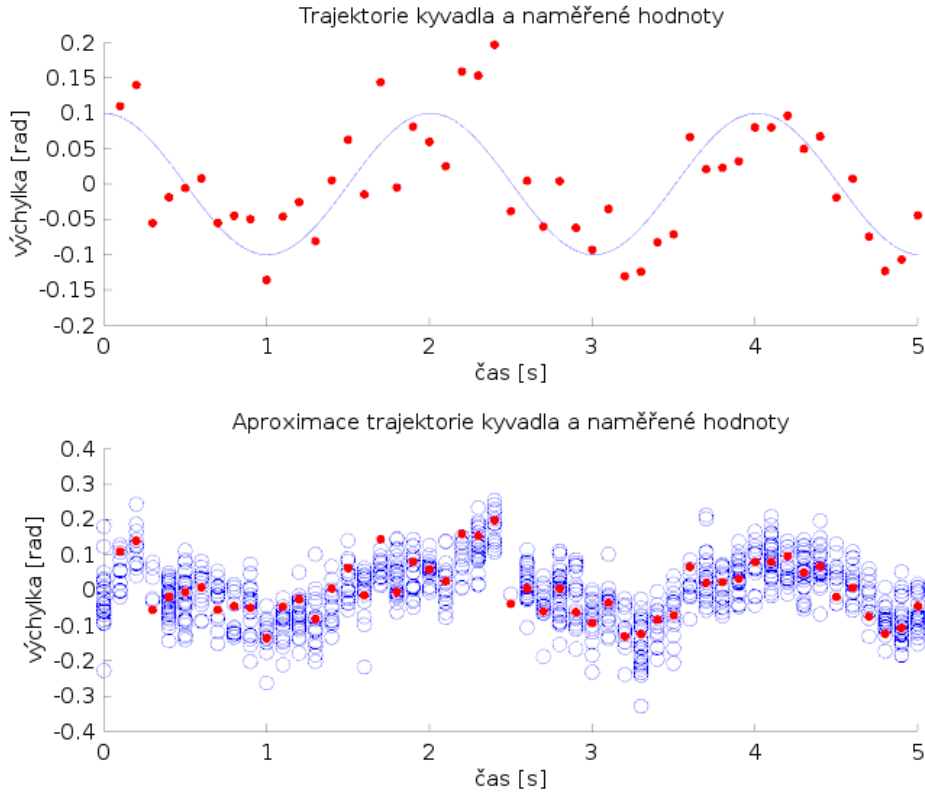
Znalost tohoto sdruženého rozdělení nám umožní přímo aplikovat algoritmus pro optimální filtraci. Zbývá zvolit vhodné návrhové hustoty. Jelikož umíme vzorkovat ze zvoleného apriorního rozdělení, volíme návrhovou hustotu  $q_0(u_0) = p_a(u_0)$ . Přechodové návrhové hustoty zvolíme  $q_k(u_k|u_{k-1}) = \mathcal{N}(u_k; u_{k-1}, 0.01)$ .

Pro řešení využijeme algoritmus 5 s parametry:

$$\begin{aligned} q_0(u_0) &= p_a(u_0) \\ w(u_0) &= 1 \\ q_k(u_k|u_{k-1}) &= \mathcal{N}(u_k; u_{k-1}, 0.01) \\ \alpha_k(u_{k-1}, u_k) &= \frac{p(y_k|u_k)p(u_k|u_{k-1})}{q_k(u_k|u_{k-1})} \end{aligned}$$

Výsledky simulace pro volbu  $N = 50, m = 50$  jsou zobrazeny na obrázku 4.

Zajímavým výstupem numerického experimentu je překvapivě slušný výsledek vzhledem k rozdílnosti skutečného a modelovaného procesu (nelineární deterministický systém x lineární gaussovský model) a nízkého počtu částic ( $N = 50$ ). Tento aspekt činí algoritmy v praxi velmi efektivní pro sledování stavu procesů s obecně neznámým vnitřním modelem. V našem případě jsme využili jen velmi jednoduchý model, který možná stačil postihnout určitou hladkost vnitřního procesu.



Obrázek 4: Sledování výchylky kyvadla. Vrchní graf znázorňuje polohu simulovaných pozorování relativně ke skutečnému stavu skrytého procesu. Spodní graf zobrazuje vzorky vniklé převzorkováním váženého souboru  $N = 50$  částic :  $(\{W_k^{1:N}, u_k^{1:N}\} \sim p(\theta|y_{1:k}))|_{k \in [1,15]}$ . Plné rudé kruhy opět znázorňují simulovaná pozorování.

### 5.2.3 Odhad tíhového zrychlení

Řešení aproximační rovnice matematického kyvadla (26) je jednoznačně určeno počátečními podmínkami a parametry úlohy. Nás v této sekci bude zajímat úloha aproximace aposteriorního rozdělení parametru evoluce skrytého procesu. Označme pro pevné parametry  $l, u_0, v_0$  řešení dané úlohy (26) pro různá  $g$ :  $\phi_g(t)$ .

Budeme se zabývat aproximací aposteriorního rozdělení parametru  $G$ . Jedná se o klasickou bayesovskou úlohu, kde aposteriorní rozdělení má tvar

$$p(g|y_{1:k}) \propto p_a(g) \prod_{n=1}^k p(y_n|g).$$

Je třeba nějak charakterizovat rozdělení  $p(y_n|g)$ . V předchozí úloze jsme pracovali s modelem pozorování  $p(y_n|u_n)$ , který se nám snadněji interpretuje. Zkusíme jej využít pro sestavení modelu i v tomto případě.

Označme  $p(y_n|g) = \int p(y_n|u_n)p(u_n|g)du_n$ , kde  $U_n$  je v tomto případě speciální pomocná n.v. repre-

zentující polohu v čase  $t_n$ . Z modelu víme, že  $u_n = \phi_g(t_n)$ . Tento jednoznačný vztah převedeme do řeči rozdělení pomocí Diracovy funkce

$$p(du_n|g_n) := \delta_{\phi_g(t_n)}(du_n).$$

Tím získáme  $p(y_n|g) = p_{Y_n|U_n}(y_n|\phi_g(t_n))$ , kde výraz na pravé straně je pomocný model pozorování kvalitativně srovnatelný s modelem v předcházející úloze (úlohu proto můžeme definovat tímto pomocným modelem pozorování).

Takto se dostaneme zpět do kontextu odhadu statického parametru, který jsme řešili v sekci 5.1 a můžeme využít stejné postupy. Tuto úlohu budeme řešit konkrétně algoritmem IS.

Experimenty na obrázcích 5 a 6 jsou provedeny se stejnými parametry,  $l, u_0, v_0$  jsou známy,

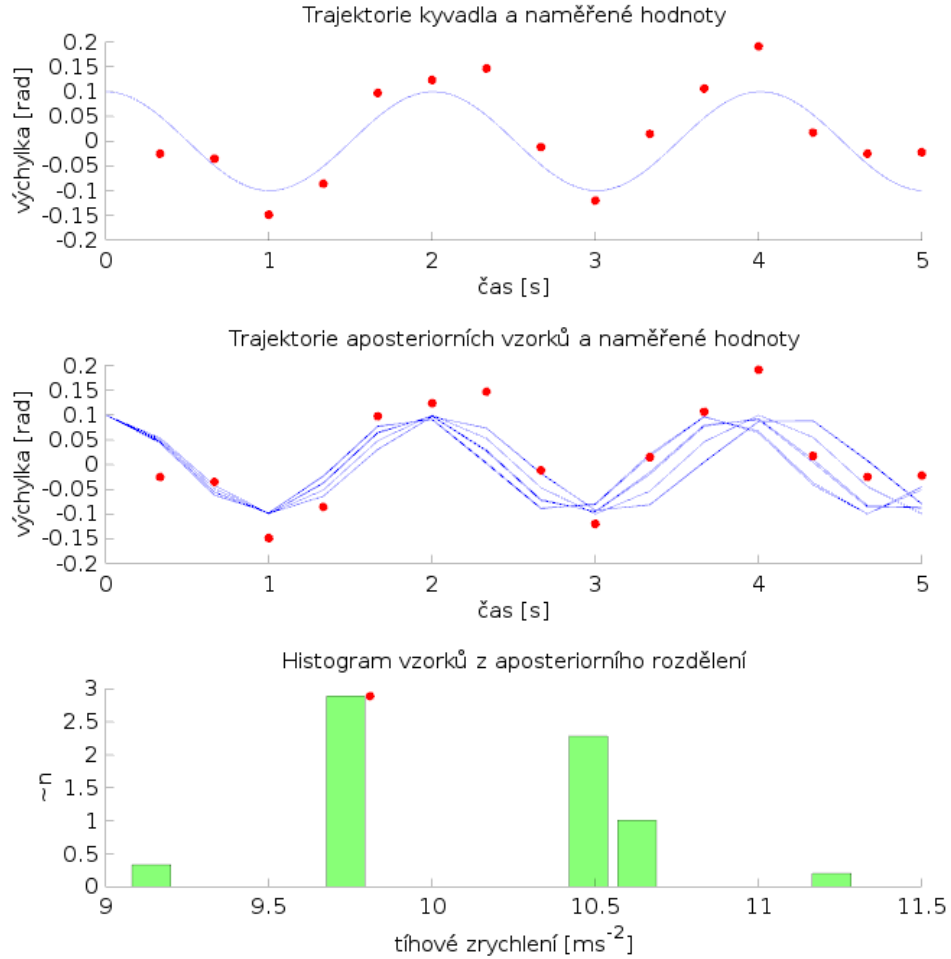
$$\begin{aligned} g^{real} &= 9.81 \\ l &= 1 \\ u_0 &= 0.1 \\ v_0 &= 0 \\ p(y_k|u_k) &= \mathcal{N}(y_k; u_k, \sigma^2) \\ p_a(g) &= \mathcal{LN}(g, 22, 100) \\ N &= 100 \\ m &= 15 \end{aligned}$$

kde  $\mathcal{LN}(g)$  zde značí hustotu log-normálního rozdělení se střední hodnotou 22 a rozptylem 100 (nejedná se o standardní zápis log-normálního rozdělení) a rozptyl modelu pozorování  $\sigma^2$  zvolíme pro následující úlohy  $\sigma^2 = 0.05; 0.1$ .

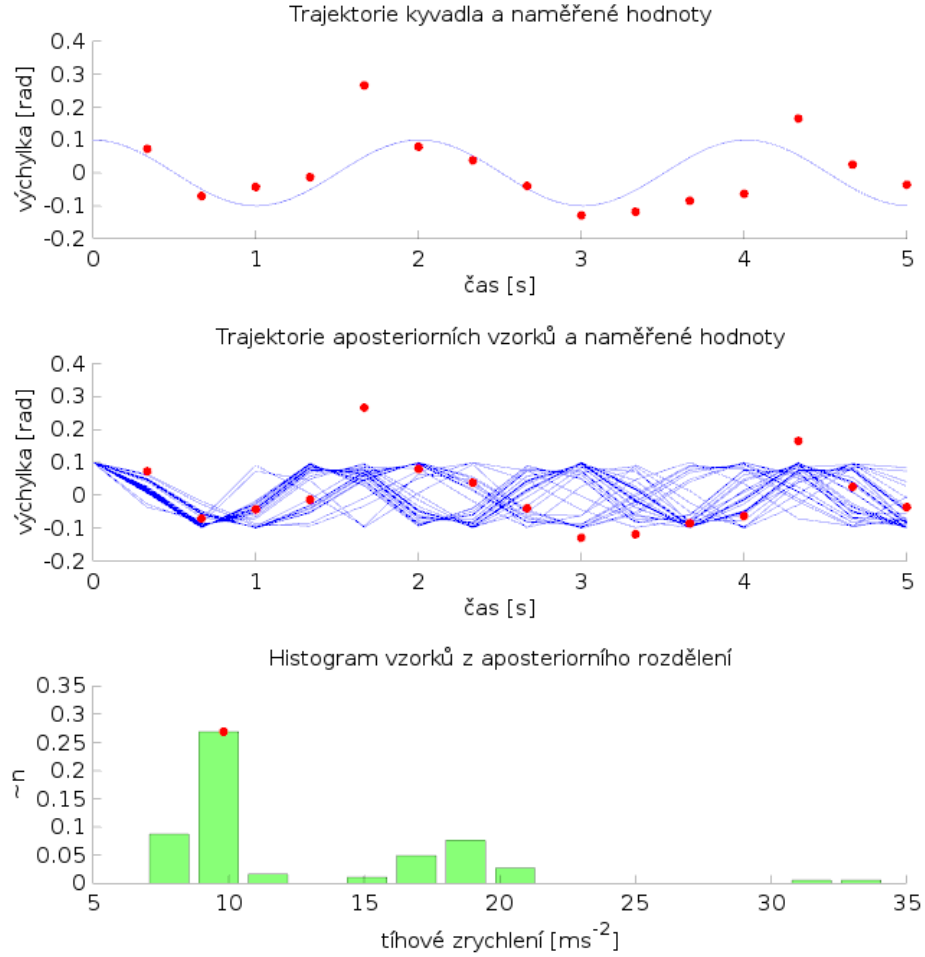
Ač jeden experiment pro každou volbu parametru  $\sigma^2$  nemá dostatečnou vypovídající hodnotu, lze odpozorovat, že v případě větší nejistoty měření se do výsledné aproximace promítne informace z apriorního rozdělení. V případě dalšího navyšování nejistoty se vnitřní dynamika začne vytrácet (náhodné chyby převáží informace o pohybu). Toto je ovšem (nejspíše) problémem samotné úlohy a nemá tedy negativní vypovídací hodnotu o použitých metodách. Na numerickém výsledku se ovšem projeví degenerace obdobně jako v sekci 5.1.1, což se ale dalo očekávat. Možným řešením tohoto problému může být využití jiného algoritmu, než je IS, čímž bychom ovšem přišli o nízkou výpočetní náročnost. V případě, kdy nás zajímá více kvalita, než rychlost, bude volba alternativního algoritmu nutností.

Můžeme si všimnout, že zmíněná úloha je speciálním případem aproximace aposteriorního rozdělení na GSSM (tedy modelu se skrytým procesem, jehož evoluce závisí na neznámém parametru).

Samotná úloha, jak je formulovaná, představuje základní metodu pro odhady parametrů deterministických procesů. Alternativní metodou, jak řešit konkrétně úlohu odhadu tíhového zrychlení za pomoci kyvadla, je získat sadu měření n.v.  $T$  reprezentující periodu kmitu, tu odhadnout a tíhové zrychlení určit ze vztahu  $g = \frac{T^2}{T}$ . Takový experiment je obvykle technicky složitější, jelikož vyžaduje speciální aparaturu, která nám bude generovat odpovídající data (např. studenta, který mačká stopky s každým průkmitem nebo čidlo, které průkmity zaznamená automaticky). Domníváme se, že popsání přístup, jehož vstupem jsou data z měření polohy, je v praxi obecně snadněji realizovatelný a mohl by být aplikován pro širší škálu úloh.



Obrázek 5: Výstup experimentu aproximace aposteriorního rozdělení neznámého parametru  $G$  pro rozptyl modelu pozorování  $\sigma^2 = 0.05$ . Vrchní graf znázorňuje skutečnou trajektorii skrytého deterministického systému a červené kruhy simulovaná pozorování. Střední graf znázorňuje některé trajektorie  $\phi_{G^i}(t)$  pro vzorky  $G^i$  z převzorkované aproximace rozdělení  $p(g|y_{1:m})$ . Červené kruhy opět znázorňují simulovaná pozorování. Spodní graf zobrazuje histogram vzorků vniklých převzorkováním váženého souboru  $\{W_m^{1:N}, u_m^{1:N}\} \sim p(g|y_{1:m})$ . Zde červený kruh značí polohu skutečného řešení.



Obrázek 6: Výstup experimentu aproximace aposteriorního rozdělení neznámého parametru  $G$  pro rozptyl modelu pozorování  $\sigma^2 = 0.1$ . Vrchní graf znázorňuje skutečnou trajektorii skrytého deterministického systému a červené kruhy simulovaná pozorování. Střední graf znázorňuje některé trajektorie  $\phi_{G^i}(t)$  pro vzorky  $G^i$  z převzorkované aproximace rozdělení  $p(g|y_{1:m})$ . Červené kruhy opět znázorňují simulovaná pozorování. Spodní graf zobrazuje histogram vzorků vniklých převzorkováním váženého souboru  $\{W_m^{1:N}, g_m^{1:N}\} \sim p(g|y_{1:m})$ . Zde červený kruh značí polohu skutečného řešení.

## 6 Závěr

V práci byly popsány základní principy sekvenčních Monte Carlo metod. Jedná se o výpočetně náročné stochastické algoritmy, které nám umožňují řešit náročné inženýrské problémy.

Sekvenční metody lze chápat jako sérii aplikací základních Monte Carlo metod (MCMC a IS). Přestože jsme popsali obecnou podobu algoritmu, konkrétní implementace jsou silně závislé na řešeném problému. Důvodem jsou prvky, které hrají roli volitelných parametrů algoritmů (hlavně návrhová rozdělení a pomocná sdružená rozdělení), na kterých závisí konvergence a náročnost celého algoritmu. Při aplikacích je proto třeba provést důslednou analýzu řešeného problému.

Hlavní oblastí aplikace SMC algoritmů je v současné době bayesovská analýza dat. Důsledně rozpracovaná je teorie pro aplikaci při řešení úlohy optimální filtrace na HMM [7, 8]. Efektivní metody existují i pro řešení úlohy optimálního vyhlazování na HMM [3, 8]. Náročnější jsou pak aplikace pro aproximaci aposteriorního rozdělení statického parametru [4, 5]. Otevřenou úlohou zůstávají aproximace aposteriorních rozdělení na general state-space modelech, které lze prozatím sekvenčními metodami řešit buď přibližně nebo s využitím speciálních vlastností konkrétních modelů [3, 7].

Pro úlohu optimální filtrace na HMM lze navrhnout výpočetně i datově efektivní algoritmy pro sekvenční výpočet aproximací hledaných rozdělení. V takových případech mohou být SMC algoritmy využity pro real-time analýzu signálů. HMM umožňují popsat pestrou paletu úloh souvisejících s analýzou časových řad. S využitím teorie rozhodování za neurčitosti můžeme formulovat úlohy řízení evolučních systémů, kdy SMC řeší podúlohu aproximace rozdělení pravděpodobnosti skrytých parametrů, které potřebujeme určit pro nalezení optimálních řídicích zásahů.

V práci jsme uvedli základní principy Monte Carlo metod, které lze využít při formulaci úlohy. Dále jsme uvedli základní principy a úlohy bayesovské statistiky a jak lze vzniklé problémy formulovat ve tvaru hledání aproximace sekvence rozdělení. Pro jejich řešení, až na výjimky, musíme využít některé z numerických metod. V aplikační části jsme vybrali několik typických úloh, které se mohou vyskytnout v praxi, a předvedli jsme výsledky, které nám poskytují zvolené SMC algoritmy.

Na práci lze přímo navázat dalším rozvojem teoretického podkladu. V sekci 4.3 je zmíněno pár možných vylepšení, které lze docela jistě dál rozvíjet. Většina vylepšení algoritmů bývá ovšem svázána s konkrétním modelem úlohy. Jelikož se jedná o algoritmy, které provádějí velké množství identických instrukcí, přirozeným vylepšením je paralelizace algoritmů. V takovém případě ovšem lze nalézt triviální schéma paralelizace přes vzorky (vytváření nových vzorků a výpočet vah jsou -většinou- operace nezávislé na zbytku souboru). Zajímavým pokračováním zdá se také býti formulace úloh mimo kontext HMM (např. pro GSSM, Markovská pole ...).

## A Vybrané partie teorie pravděpodobnosti

V této sekci uvádíme některé základní poznatky z teorie pravděpodobnosti a matematické statistiky, na které se v textu odvoláváme. Jedná se o tvrzení, která se přímo nehodí jako součást hlavního textu.

Tvrzení jsou uváděna bez komentářů. Zdrojem je skriptum [16].

### A.1 Vybrané definice

**Definice A.1.** (Pravděpodobnostní míra)

Pravděpodobnostní mírou  $P$  na  $\sigma$ -algebře  $\mathcal{A} \subset 2^\Omega$  nazveme konečnou míru na  $\mathcal{A}$ , pro kterou platí

$$\cdot P(\Omega) = 1$$

**Definice A.2.** (Pravděpodobnostní prostor)

Pravděpodobnostním prostorem nazveme trojici  $(\Omega, \mathcal{A}, P)$ , kde

- $\Omega$  nazýváme prostor jevů
- $\mathcal{A}$  je  $\sigma$ -algebrou na  $\Omega$  ( $(\Omega, \mathcal{A})$  tvoří měřitelný prostor)
- $P$  je pravděpodobnostní míra na  $\mathcal{A}$

**Definice A.3.** (Náhodná veličina)

Bud'  $(\Omega, \mathcal{A}, P)$  pravděpodobnostní prostor a  $(E, \mathcal{E})$  měřitelný prostor.

Měřitelné zobrazení  $X : \Omega \rightarrow E$  nazveme náhodnou veličinou.

Obor hodnot n.v.  $X$  budeme značit  $R(X)$ .

*Poznámka A.4.* (Indukovaná míra)

Náhodná veličina  $X : \Omega \rightarrow E$  indukuje pravděpodobnostní míru  $\pi$  definovanou :

$$\pi(e) = P(X^{-1}(e)); \forall e \in \mathcal{E}$$

**Definice A.5.** (Absolutní spojitost měr)

Bud'  $(\Omega, \Sigma)$  měřitelný prostor a  $\eta, \mu$  míry na  $(\Omega, \Sigma)$ .

Řekneme, že míra  $\eta$  je absolutně spojitá vůči míře  $\mu$ , platí-li  $\forall A \in \Sigma : \mu(A) = 0 \Rightarrow \eta(A) = 0$  a značíme  $\eta \ll \mu$ .

**Věta A.6.** (Radon-Nikodym)

Bud'  $(\Omega, \Sigma)$  měřitelný prostor a  $\eta, \mu$   $\sigma$ -konečné míry na  $(\Omega, \Sigma)$ ;  $\eta \ll \mu$ .

Pak existuje měřitelná funkce  $f : \Omega \rightarrow [0, \infty)$  taková, že pro každou  $A \in \Sigma$  platí:

$$\eta(A) = \int_A f d\mu$$

Funkci  $f$  nazveme Radonovou-Nikodýmovou derivací a značíme  $\frac{d\eta}{d\mu}$ .

Funkce  $f$  je určena jednoznačně skoro všude v  $\mu$ .



**Definice A.7.** (Hustota pravděpodobnosti)

Bud'  $(\Omega, \mathcal{A}, P)$  pravděpodobnostní prostor a  $X : \Omega \rightarrow E$  náhodná veličina indukující míru  $\pi$ . Označme  $dx$  Borelovu míru na  $E$ .

Existuje-li Radonova-Nikodýmova derivace  $\frac{d\pi}{dx}$ , nazveme ji hustotou pravděpodobnosti (probability density function - pdf).

**Definice A.8.** (Kumulativní distribuční funkce)

Bud'  $(\Omega, \mathcal{A}, P)$  pravděpodobnostní prostor a  $X : (\Omega, \mathcal{A}, P) \rightarrow (\mathbb{R}^n, \mathcal{B})$ , kde  $\mathcal{B}$  je Borelovská  $\sigma$ -algebra na  $\mathbb{R}^n$ , náhodná veličina indukující míru  $\pi(dx)$ .

Kumulativní distribuční funkcí (cdf) nazveme funkci  $F : \mathbb{R}^n \rightarrow \langle 0, 1 \rangle$  definovanou:

$$F(x_1, x_2, \dots, x_n) = \left( \int_{-\infty}^{x_1} \dots \int_{-\infty}^{x_n} \pi(dx_1, \dots, dx_n) \right)$$

**A.2 Konvergenční věty****Definice A.9.** (Konvergence skoro jistě)

Bud'  $X_n, n = 1, 2, \dots$  náhodné veličiny na prostoru  $(\Omega, \mathcal{A}, P)$ . Pokud pravděpodobnost jevu

$$a := \{\omega; X_n \rightarrow X\}$$

je  $Pr(a) = 1$ . Řekneme, že posloupnost  $X_n$  konverguje téměř jistě k  $X$ .

Značíme

$$X_n \xrightarrow{a.s.} X$$

**Věta A.10.** (Silný zákon velkých čísel)

Bud'  $X_n$  řada integrovatelných ( $\mathbb{E}|X_i| < \infty$ ) náhodných veličin i.i.d se střední hodnotou  $\mathbb{E}X_i = m$ , pak

$$\frac{1}{n} \sum_{i=1}^n X_i \xrightarrow{a.s.} m$$

**Definice A.11.** (Konvergence v distribuci)

Řekneme, že řada náhodných veličin  $X_n$  konverguje v distribuci k  $X$ , pokud pro příslušné c.d.f.  $F_n, F$  platí:

$$F_n(x) \rightarrow F(x) \quad \text{ve všech bodech spojitosti } F(x).$$

Značíme

$$X_n \Rightarrow X$$

**Věta A.12.** (*Centrální limitní věta*)

*Jsou-li náhodné veličiny  $X_i$  i.i.d. se střední hodnotou  $\mathbb{E}X_i = m$  a rozptylem  $DX_i = \sigma^2$ , pak platí*

$$\frac{1}{\sigma\sqrt{n}} \sum_{i=1}^n (X_i - m) \Rightarrow \mathcal{N}(0, 1)$$

## Reference

- [1] Christophe Andrieu, Nando De Freitas, Arnaud Doucet, and Michael I Jordan. An introduction to MCMC for machine learning. *Machine learning*, 50(1-2):5–43, 2003.
- [2] James O. Berger. *Statistical decision theory and Bayesian analysis*. Springer, 1985.
- [3] Carlos Carvalho, Michael S Johannes, Hedibert F Lopes, and Nick Polson. Particle learning and smoothing. *Statistical Science*, 25(1):88–106, 2010.
- [4] Nicolas Chopin. A sequential particle filter method for static models. *Biometrika*, 89(3):539–552, 2002.
- [5] Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. Sequential monte carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):411–436, 2006.
- [6] Randal Douc and Olivier Cappé. Comparison of resampling schemes for particle filtering. In *Image and Signal Processing and Analysis, 2005. ISPA 2005. Proceedings of the 4th International Symposium on*, pages 64–69. IEEE, 2005.
- [7] Arnaud Doucet, Nando de Freitas, and Neil Gordon. *Sequential Monte Carlo Methods in Practice*. Springer Science & Business Media, 2001.
- [8] Arnaud Doucet and Adam M Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of Nonlinear Filtering*, 12:656–704, 2009.
- [9] Ramsey Faragher et al. Understanding the basis of the kalman filter via a simple and intuitive derivation. *IEEE Signal processing magazine*, 29(5):128–132, 2012.
- [10] David Freedman. On the bernstein-von mises theorem with infinite-dimensional parameters. *Annals of Statistics*, pages 1119–1140, 1999.
- [11] Andrew Gelman and Cosma Rohilla Shalizi. Philosophy and the practice of bayesian statistics. *British Journal of Mathematical and Statistical Psychology*, 66(1):8–38, 2013.
- [12] Jun S Liu. *Monte Carlo strategies in scientific computing*. Springer Science & Business Media, 2008.
- [13] Hušková M. *Bayesovské metody*. Univerzita Karlova, 1985.
- [14] R. Briš M. Béréš. *Metoda Monte Carlo jako nástroj pro řešení inženýrských problémů, Bakalářská práce*. Vysoká Škola Báňská-Technická Univerzita Ostrava, FEI, 2013.
- [15] Reinaldo Marques and Geir Storvik. Particle move-reweighting schemes for online Bayesian inference.
- [16] Don L. McLeish. STAT 901:Probability. Skriptum, online:<http://sas.uwaterloo.ca/dlmcleis/s901/>, 2003.
- [17] Nicholas Metropolis. *The Begginings of Monte Carlo method*, pages 125–130. 15. Los Alamos Science, 1997.

- [18] Adrian FM Smith and Alan E Gelfand. Bayesian statistics without tears: a sampling–resampling perspective. *The American Statistician*, 46(2):84–88, 1992.
- [19] B. Walsh. Markov chain monte carlo and gibbs sampling, 2004.
- [20] Greg Welch and Gary Bishop. An introduction to the kalman filter, 1995.